Multi-Cell Radio Resource Management
for future cellular systems

Mohammad Abaii

Submitted for the Degree of
Doctor of Philosophy
from the
University of Surrey

Mobile Communications Research Group
Centre for Communication Systems Research
Faculty of Engineering and Physical Sciences
University of Surrey
Guildford, Surrey GU2 7XH, UK

July 2009

© Mohammad Abaii 2009
Abstract

Future mobile communications systems will be designed to support a wide range of data rates with complex and conflicting quality of service requirements. It is becoming more challenging to optimize radio resource management and maximize the system capacity whilst meeting the required quality of service from users' point of view. Traditional techniques have approached this problem by mainly focusing on resources within a cell and to large extent ignoring effects of multi-cell architecture leading to non-uniform and unstable capacity across the network.

This thesis first investigates the potential performance improvements obtained by developing novel distributed scheduling algorithms thereby highlighting the shortcomings of conventional single-cell scheduling techniques in a multi-cell system. It was found that distributed scheduling can achieve superior performance (up to 30% increased cell throughput) compared to conventional one in low/medium system loading. However, there is little advantage in case of heavily loaded system.

The main achievement in this thesis is addressing this problem by proposal of a novel technique called Load Matrix, setting a new direction for future research on resource scheduling strategies in a multi-cell system. LM facilitates joint management of interference within and between cells for efficient allocation of radio resources. Simulation results provided show significant improvement in the resource utilization and overall network performance. Using LM technique, the average cell throughput can be increased between 30% to 50%. Results also show that maintaining cell interference within a margin as opposed to a hard target, can significantly improve resource utilization over time (longevity) and over the cells (uniformity). The thesis also compares the effect of ideal LM with practical and implementable versions with channel gain errors, information delay, and reducing LM database to adjacent cells. The conclusion was interesting as the performance degradation in practical LM compared to ideal LM was found to be negligible.

**Key words:** Coordinated Multi-Point transmission (CoMP), Radio resource management (RRM), radio resource allocation, resource scheduling, packet scheduling, Interference management
Acknowledgments

I sincerely thank my parents and my family for their consistent support throughout my study. Also my special thanks to Professor Rahim Tafazolli for his guidance and advice, and to my colleagues at mobile communications research group in CCSR for their suggestions and support during my research.
Contents

Abstract .......................................................................................................................... ii
Acknowledgments ......................................................................................................... iii
Contents .......................................................................................................................... iv
List of Figures ................................................................................................................ vii
List of Tables .................................................................................................................. xi
Glossary of Terms .......................................................................................................... xii

1 Introduction .................................................................................................................. 1
   1.1 Motivation and Objectives ....................................................................................... 1
   1.2 Contributions and Achievements ............................................................................. 3
      1.2.1 Two new distributed scheduling algorithms .................................................. 3
      1.2.2 Identified the shortcomings of traditional scheduling (single-cell based) ......... 4
      1.2.3 Introduced Load Matrix technique (multi-cell based) ...................................... 4
      1.2.4 LM full analysis under ideal and practical assumptions .................................... 4
      1.2.5 Stability of system performance achieved by LM technique ............................ 5
   1.3 Thesis Outline ......................................................................................................... 5
   1.4 Summary .................................................................................................................. 7

2 Scheduling in Uplink .................................................................................................. 9
   2.1 Introduction ............................................................................................................. 9
   2.2 Nokia’s NodeB Scheduling Algorithm .................................................................. 15
      2.2.1 Model Structure ............................................................................................. 15
      2.2.2 Algorithm Description .................................................................................... 16
   2.3 Qualcomm’s NodeB Scheduling Algorithm ............................................................. 17
      2.3.1 Model Structure ............................................................................................. 17
      2.3.2 Algorithm Description .................................................................................... 18
   2.4 Virtually Centralised Scheduler (VCUP) ................................................................. 19
      2.4.1 Model structure .............................................................................................. 19
      2.4.2 Algorithm Description .................................................................................... 21
   2.5 Summary .................................................................................................................. 23

3 Developing Distributed Scheduling Algorithms ...................................................... 24
### Contents

3.1 Introduction ....................................................................................................................... 24

3.2 Distributed Scheduling UniS Algo-1 ....................................................................................... 25

3.2.1 Parameter definition and Initialization........................................................................ 25

3.2.2 Algo-1 description ....................................................................................................... 26

3.3 Distributed Scheduling UniS Algo-2 ....................................................................................... 27

3.3.1 Parameter definition and Initialization ........................................................................ 27

3.3.2 Algo-2 description ....................................................................................................... 28

3.4 Summary ........................................................................................................................... 30

4 System level simulations ............................................................................................. 31

4.1 Introduction ....................................................................................................................... 31

4.2 Overview of the Simulation Platform ............................................................................... 31

4.2.1 Simulation Topology .................................................................................................. 31

4.2.2 Main Features ............................................................................................................. 32

4.3 Simulation Validation ........................................................................................................ 32

4.3.1 Simulation Methodology ............................................................................................ 32

4.3.2 RoT Performance ........................................................................................................ 33

4.3.3 UE Throughput Performance ...................................................................................... 34

4.4 Scheduling Simulation Results .......................................................................................... 35

4.4.1 Objectives ................................................................................................................... 35

4.4.2 Simulation Cases ......................................................................................................... 35

4.4.3 Analysis ...................................................................................................................... 36

4.5 Summary ........................................................................................................................... 42

5 Importance of Intercell Interference Control...................................................................... 44

5.1 RoT as resource utilization indicator ................................................................................. 44

5.2 RoT fluctuation in time ..................................................................................................... 46

5.3 Summary ........................................................................................................................... 49

6 Load Matrix Scheduling Technique ............................................................................. 51

6.1 The concept and the constraints ....................................................................................... 51

6.2 Load Matrix definition ...................................................................................................... 55

6.3 Simulation Results ............................................................................................................ 60

6.4 LM - Priority functions combination ................................................................................. 68

6.5 Summary ........................................................................................................................... 72

7 Assessment on Load Matrix impairments .................................................................... 74

7.1 Introduction ....................................................................................................................... 74
## Contents

7.2 Load Matrix projection ................................................................. 75
7.3 Load Matrix with channel estimation/measurement error ............... 80
7.4 Load Matrix restricted to neighbour elements .............................. 83
7.5 Load Matrix with additional delay .............................................. 88
7.6 Summary ..................................................................................... 91

8 Stability of Load Matrix ................................................................. 93
  8.1 Stability in interference control .................................................. 94
  8.2 Stability in Throughput .............................................................. 97
  8.3 Summary ..................................................................................... 99

9 Conclusion and Future Work ......................................................... 101
  9.1 Conclusion ................................................................................ 101
  9.2 Future Work ............................................................................ 103

Publications ....................................................................................... 105

Award ............................................................................................... 105

Bibliography ..................................................................................... 106

Appendix A: Simulation platform ....................................................... 110
  A.1 Introduction ............................................................................. 110
  A.2 Evaluation summary of VCUP algorithm ................................. 110
    A.2.1 CM2 (VCUP1) ................................................................. 110
    A.2.2 CM3 (VCUP2) ................................................................. 112
  A.3 Hotspot Modelling ................................................................. 114
  A.4 System level Simulations .......................................................... 115
    A.4.1 Simulation Results for Cases without Hot Spots ................. 118
      A.4.1.1 Cell Load: 10 UEs per Cell ..................................... 118
      A.4.1.2 Cell Load: 15 UEs per Cell ..................................... 123
    A.4.2 Simulation Results for Cases with Hot Spots ..................... 128
      A.4.2.1 Hot Spot Simulation Case A ................................. 128
      A.4.2.2 Hot Spot Simulation Case B ................................. 133
    A.4.3 Summary ......................................................................... 139
List of Figures

Figure 2-1: HSUPA Centralized scheduling ................................................................. 12
Figure 2-2: HSUPA Decentralized scheduling............................................................... 12
Figure 2-3: Definition of the TFC pointers .................................................................. 13
Figure 2-4: Timing diagram for request, scheduling and assignment in decentralized scheduling 14
Figure 2-5: Decision regions for UEs partitioning at Node B ........................................... 15
Figure 2-6: First stage of rate selection (same as in TFC selection Release 99) .............. 20
Figure 2-7: Second stage of rate selection in VCUP (using CM) .................................... 20
Figure 3-1: Distributed scheduling Algo-1 flowchart ..................................................... 26
Figure 3-2: Distributed scheduling Algo-2 flowchart ..................................................... 29
Figure 4-1: Network deployment .................................................................................... 32
Figure 4-2: User distribution snapshot (10 UEs per cell) ............................................. 33
Figure 4-3: RoT vs. number of UEs per cell ................................................................. 34
Figure 4-4: User throughput vs. downlink path loss of 3GPP release-99 system ............ 34
Figure 4-5: User throughput vs. downlink path loss from our simulation platform ......... 35
Figure 4-6: PDF of average Cell RoT ............................................................................ 37
Figure 4-7: CDF of average Cell RoT ........................................................................... 37
Figure 4-8: PDF of system total queue size ................................................................. 38
Figure 4-9: CDF of system total queue size ................................................................. 38
Figure 4-10: PDF of packet delay in number of TTIs ...................................................... 39
Figure 4-11: CDF of packet delay in number of TTIs ...................................................... 39
Figure 4-12: OTA throughput vs. distance from cell site .............................................. 40
Figure 4-13: Service throughput vs. distance from cell site ......................................... 41
Figure 4-14: Buffer occupancy vs. distance from cell site ........................................... 41
Figure 4-15: Mean packet delay vs. distance from cell site ......................................... 42
Figure 5-1: Resource Utilization interpretation of RoT .................................................. 45
Figure 5-2: Typical RoT fluctuation (in time) in conventional scheduling due to intercell interference ................................................................. 46
Figure 5-3: Extreme scenario with identical cells ......................................................... 47
Figure 5-4: RoT fluctuation over time (20 second) in extreme scenario ......................... 48
Figure 5-5: RoT fluctuation in extreme scenario (1 second observation) ....................... 49
Figure 6-1: Centralized LM scheduling in a 3G LTE system ......................................... 55
Figure 6-2: PDF of RoT (inter-cell margin effect) ......................................................... 61
Figure 6-3: CDF of RoT (inter-cell margin effect) ......................................................... 62
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6-4</td>
<td>PDF of RoT (intra-cell margin effect)</td>
<td>63</td>
</tr>
<tr>
<td>6-5</td>
<td>CDF of RoT (intra-cell margin effect)</td>
<td>63</td>
</tr>
<tr>
<td>6-6</td>
<td>PDF of RoT (best three combinations)</td>
<td>64</td>
</tr>
<tr>
<td>6-7</td>
<td>CDF of RoT (best three combinations)</td>
<td>65</td>
</tr>
<tr>
<td>6-8</td>
<td>Average service throughput versus distance</td>
<td>65</td>
</tr>
<tr>
<td>6-9</td>
<td>Histogram of Packet delay</td>
<td>66</td>
</tr>
<tr>
<td>6-10</td>
<td>CDF of packet delay</td>
<td>67</td>
</tr>
<tr>
<td>6-11</td>
<td>RoT fluctuation with LM scheduler in the same cell/time as shown in Figure 5-2</td>
<td>67</td>
</tr>
<tr>
<td>6-12</td>
<td>CDF of RoT for LM-enhanced schedulers</td>
<td>70</td>
</tr>
<tr>
<td>6-13</td>
<td>Average cell throughput for LM-enhanced schedulers</td>
<td>70</td>
</tr>
<tr>
<td>6-14</td>
<td>Throughput over distance - effect of $W$ size in GPF</td>
<td>71</td>
</tr>
<tr>
<td>6-15</td>
<td>Throughput over distance (range fairness) comparison</td>
<td>72</td>
</tr>
<tr>
<td>7-1</td>
<td>load element categories used in Load Matrix terminology and projection</td>
<td>76</td>
</tr>
<tr>
<td>7-2</td>
<td>A typical snapshot of Load Matrix projection in a cell for significant elements (bigger than $\text{RoT}_{target}/1000$)</td>
<td>77</td>
</tr>
<tr>
<td>7-3</td>
<td>Load Vector projection for cell no. 12 (best case)</td>
<td>78</td>
</tr>
<tr>
<td>7-4</td>
<td>Load Vector projection for cell no. 1 (worst case)</td>
<td>79</td>
</tr>
<tr>
<td>7-5</td>
<td>Load Matrix projection after implementing LM scheduler</td>
<td>79</td>
</tr>
<tr>
<td>7-6</td>
<td>PDF of RoT in LM scheduling with channel error</td>
<td>81</td>
</tr>
<tr>
<td>7-7</td>
<td>CDF of RoT in LM scheduling with channel error</td>
<td>81</td>
</tr>
<tr>
<td>7-8</td>
<td>Cell throughput versus distance with and without channel error</td>
<td>82</td>
</tr>
<tr>
<td>7-9</td>
<td>CDF of packet delay with and without channel error</td>
<td>83</td>
</tr>
<tr>
<td>7-10</td>
<td>PDF of RoT in LM scheduling with restriction to neighbours and with channel errors</td>
<td>85</td>
</tr>
<tr>
<td>7-11</td>
<td>CDF of RoT in LM scheduling with restriction to neighbours with channel errors</td>
<td>85</td>
</tr>
<tr>
<td>7-12</td>
<td>Cell throughput versus distance with and without channel error</td>
<td>86</td>
</tr>
<tr>
<td>7-13</td>
<td>CDF of packet delay in LM scheduling with restriction to neighbours with and without channel error</td>
<td>87</td>
</tr>
<tr>
<td>7-14</td>
<td>PDF of RoT in LM scheduling with additional delay</td>
<td>89</td>
</tr>
<tr>
<td>7-15</td>
<td>CDF of RoT in LM scheduling with additional delay</td>
<td>90</td>
</tr>
<tr>
<td>7-16</td>
<td>Cell throughput versus distance for LM scheduling with additional delay</td>
<td>90</td>
</tr>
<tr>
<td>7-17</td>
<td>CDF of packet delay in LM scheduling with additional delay</td>
<td>91</td>
</tr>
<tr>
<td>8-1</td>
<td>Cell RoT Stability comparison (best cell and worst cell)</td>
<td>95</td>
</tr>
<tr>
<td>8-2</td>
<td>Qualcomm 3D RoT plot</td>
<td>95</td>
</tr>
<tr>
<td>8-3</td>
<td>Load Matrix 3D RoT plot</td>
<td>96</td>
</tr>
<tr>
<td>8-4</td>
<td>Cell Throughput Stability (best cell and worst cell)</td>
<td>98</td>
</tr>
<tr>
<td>8-5</td>
<td>Qualcomm 3D Throughput Stability plot</td>
<td>98</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>A-1</td>
<td>Cell layout with hot spots</td>
<td></td>
</tr>
<tr>
<td>A-2</td>
<td>User distribution (10 UEs per cell, hot spot case B)</td>
<td></td>
</tr>
<tr>
<td>A-3</td>
<td>PDF of RoT (10 UEs per cell)</td>
<td></td>
</tr>
<tr>
<td>A-4</td>
<td>CDF of RoT (10 UEs per cell)</td>
<td></td>
</tr>
<tr>
<td>A-5</td>
<td>PDF of total queue size (10 UEs per cell)</td>
<td></td>
</tr>
<tr>
<td>A-6</td>
<td>CDF of total queue size (10 UEs per cell)</td>
<td></td>
</tr>
<tr>
<td>A-7</td>
<td>PDF of packet delay (10 UEs per cell)</td>
<td></td>
</tr>
<tr>
<td>A-8</td>
<td>CDF of packet delay (10 UEs per cell)</td>
<td></td>
</tr>
<tr>
<td>A-9</td>
<td>OTA throughput vs. distance from cell site (10 UEs per cell)</td>
<td></td>
</tr>
<tr>
<td>A-10</td>
<td>Service throughput vs. distance from cell site (10 UEs per cell)</td>
<td></td>
</tr>
<tr>
<td>A-11</td>
<td>Buffer occupancy vs. distance from cell site (10 UEs per cell)</td>
<td></td>
</tr>
<tr>
<td>A-12</td>
<td>Mean packet delay vs. distance from cell site (10 UEs per cell)</td>
<td></td>
</tr>
<tr>
<td>A-13</td>
<td>PDF of RoT (15 UEs per cell)</td>
<td></td>
</tr>
<tr>
<td>A-14</td>
<td>CDF of RoT (15 UEs per cell)</td>
<td></td>
</tr>
<tr>
<td>A-15</td>
<td>PDF of total queue size (15 UEs per cell)</td>
<td></td>
</tr>
<tr>
<td>A-16</td>
<td>CDF of total queue size (15 UEs per cell)</td>
<td></td>
</tr>
<tr>
<td>A-17</td>
<td>PDF of packet delay (15 UEs per cell)</td>
<td></td>
</tr>
<tr>
<td>A-18</td>
<td>CDF of packet delay (15 UEs per cell)</td>
<td></td>
</tr>
<tr>
<td>A-19</td>
<td>OTA throughput vs. distance from cell site (15 UEs per cell)</td>
<td></td>
</tr>
<tr>
<td>A-20</td>
<td>Service throughput vs. distance from cell site (15 UEs per cell)</td>
<td></td>
</tr>
<tr>
<td>A-21</td>
<td>Buffer occupancy vs. distance from cell site (15 UEs per cell)</td>
<td></td>
</tr>
<tr>
<td>A-22</td>
<td>Mean packet delay vs. distance from cell site (15 UEs per cell)</td>
<td></td>
</tr>
<tr>
<td>A-23</td>
<td>PDF of RoT (10 UEs per cell, hot spot case A)</td>
<td></td>
</tr>
<tr>
<td>A-24</td>
<td>CDF of RoT (10 UEs per cell, hot spot case A)</td>
<td></td>
</tr>
<tr>
<td>A-25</td>
<td>PDF of total queue size (10 UEs per cell, hot spot case A)</td>
<td></td>
</tr>
<tr>
<td>A-26</td>
<td>CDF of total queue size (10 UEs per cell, hot spot case A)</td>
<td></td>
</tr>
<tr>
<td>A-27</td>
<td>PDF of packet delay (10 UEs per cell, hot spot case A)</td>
<td></td>
</tr>
<tr>
<td>A-28</td>
<td>CDF of packet delay (10 UEs per cell, hot spot case A)</td>
<td></td>
</tr>
<tr>
<td>A-29</td>
<td>OTA throughput vs. distance from cell site (10 UEs per cell, hot spot case A)</td>
<td></td>
</tr>
<tr>
<td>A-30</td>
<td>Service throughput vs. distance from cell site (10 UEs per cell, hot spot case A)</td>
<td></td>
</tr>
<tr>
<td>A-31</td>
<td>Buffer occupancy vs. distance from cell site (10 UEs per cell, hot spot case A)</td>
<td></td>
</tr>
<tr>
<td>A-32</td>
<td>Mean packet delay vs. distance from cell site (10 UEs per cell, hot spot case A)</td>
<td></td>
</tr>
<tr>
<td>A-33</td>
<td>PDF of RoT (10 UEs per cell, hot spot case B)</td>
<td></td>
</tr>
<tr>
<td>A-34</td>
<td>CDF of RoT (10 UEs per cell, hot spot case B)</td>
<td></td>
</tr>
<tr>
<td>A-35</td>
<td>PDF of total queue size (10 UEs per cell, hot spot case B)</td>
<td></td>
</tr>
<tr>
<td>A-36</td>
<td>CDF of total queue size (10 UEs per cell, hot spot case B)</td>
<td></td>
</tr>
</tbody>
</table>
List of Figures

Figure A-37: PDF of packet delay (10 UEs per cell, hot spot case B) ......................................... 135
Figure A-38: CDF of packet delay (10 UEs per cell, hot spot case B) ......................................... 136
Figure A-39: OTA throughput vs. distance from cell site (10 UEs per cell, hot spot case B) .... 136
Figure A-40: Service throughput vs. distance from cell site (10 UEs per cell, hot spot case B). 137
Figure A-41: Buffer occupancy vs. distance from cell site (10 UEs per cell, hot spot case B)... 137
Figure A-42: Mean packet delay vs. distance from cell site (10 UEs per cell, hot spot case B)... 138
List of Tables

Table 6.1: SCRAP to MCKP parameter mapping .............................................................. 53
Table 6.2: LM Technique; summary ................................................................................ 58
Table 6.3: Simulation parameters summary ....................................................................... 60
Table A.1: Simulation Parameters .................................................................................... 117
Table A.2: Results summary for hot spot performance (case A) .................................. 133
Table A.3: Results summary for hot spot performance (case B) ..................................... 138
# Glossary of Terms

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>3GPP</td>
<td>3rd Generation Partnership Project</td>
</tr>
<tr>
<td>B3G</td>
<td>Beyond 3rd Generation</td>
</tr>
<tr>
<td>BS</td>
<td>Base Station</td>
</tr>
<tr>
<td>BLER</td>
<td>Block Error Rate</td>
</tr>
<tr>
<td>CB</td>
<td>Comparative Buffer</td>
</tr>
<tr>
<td>CC</td>
<td>Capacity Check</td>
</tr>
<tr>
<td>CDF</td>
<td>Cumulative Distribution Function</td>
</tr>
<tr>
<td>CI</td>
<td>Capacity Indicator</td>
</tr>
<tr>
<td>CM</td>
<td>Comparative Metric</td>
</tr>
<tr>
<td>CoMP</td>
<td>Coordinated Multi-Point</td>
</tr>
<tr>
<td>CRS</td>
<td>Candidate Rate Set</td>
</tr>
<tr>
<td>FDD</td>
<td>Frequency Division Duplex</td>
</tr>
<tr>
<td>GPF</td>
<td>Global Proportional Fair</td>
</tr>
<tr>
<td>GPP</td>
<td>Global Proportional Priority</td>
</tr>
<tr>
<td>HSDPA</td>
<td>High Speed Downlink Packet Access</td>
</tr>
<tr>
<td>HSUPA</td>
<td>High Speed Uplink Packet Access</td>
</tr>
<tr>
<td>LM</td>
<td>Load Matrix</td>
</tr>
<tr>
<td>LTE</td>
<td>Long Term Evolution</td>
</tr>
<tr>
<td>MAC</td>
<td>Medium Access Control</td>
</tr>
<tr>
<td>MCKP</td>
<td>Multi-Choice Knapsack Problem</td>
</tr>
<tr>
<td>MCRAP</td>
<td>Multi-Cell Radio Allocation Problem</td>
</tr>
<tr>
<td>MMSE</td>
<td>Minimum Mean Square Error</td>
</tr>
<tr>
<td>MUD</td>
<td>Multi-User Detection</td>
</tr>
<tr>
<td>NodeB</td>
<td>Base station in UMTS</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Definition</td>
</tr>
<tr>
<td>--------------</td>
<td>------------</td>
</tr>
<tr>
<td>NQ</td>
<td>Normalized Queue</td>
</tr>
<tr>
<td>NR</td>
<td>Noise Rise</td>
</tr>
<tr>
<td>NRT</td>
<td>Noise Rise Target</td>
</tr>
<tr>
<td>NP</td>
<td>Non deterministic Polynomial time</td>
</tr>
<tr>
<td>OTA</td>
<td>Over The Air</td>
</tr>
<tr>
<td>PDF</td>
<td>Probability Distribution Function</td>
</tr>
<tr>
<td>PF</td>
<td>Proportional Fair</td>
</tr>
<tr>
<td>RNC</td>
<td>Radio Network Controller</td>
</tr>
<tr>
<td>RoT</td>
<td>Rise over Thermal noise</td>
</tr>
<tr>
<td>RR</td>
<td>Round Robin</td>
</tr>
<tr>
<td>RRM</td>
<td>Radio Resource Management</td>
</tr>
<tr>
<td>RUF</td>
<td>Resource Utilization Factor</td>
</tr>
<tr>
<td>SB</td>
<td>Score Based</td>
</tr>
<tr>
<td>SCRAP</td>
<td>Single Cell Radio Allocation Problem</td>
</tr>
<tr>
<td>SHO</td>
<td>Soft Hand Over</td>
</tr>
<tr>
<td>SIR</td>
<td>Signal to Interference Ratio</td>
</tr>
<tr>
<td>SINR</td>
<td>Signal to Interference and Noise Ratio</td>
</tr>
<tr>
<td>TDMA</td>
<td>Time Division Multiple Access</td>
</tr>
<tr>
<td>TFC</td>
<td>Transport Format Combination</td>
</tr>
<tr>
<td>TFCS</td>
<td>Transport Format Combination Subset</td>
</tr>
<tr>
<td>TTI</td>
<td>Transmission Time Interval</td>
</tr>
<tr>
<td>UE</td>
<td>User Equipment</td>
</tr>
<tr>
<td>UMTS</td>
<td>Universal Mobile Telecommunication System</td>
</tr>
<tr>
<td>UT</td>
<td>User Terminal</td>
</tr>
<tr>
<td>UTRA</td>
<td>Universal Terrestrial Radio Access</td>
</tr>
<tr>
<td>VCUP</td>
<td>Virtually Centralized Uplink Scheduler</td>
</tr>
</tbody>
</table>
1 Introduction

1.1 Motivation and Objectives

Mobile cellular systems are facing new challenges since new applications have become more and more sophisticated and widely used. Internet growth in the past decade has changed our way of life. People, especially youth have become internet generation with growing desire for instant access to information. New services and applications have changed our work environment and the way we spend our time. Almost all circuit switched applications are replacing with more complicated packet switched ones and the demand for broadband and higher bandwidth communications have been increased substantially. Users expect ubiquitous communication anywhere anytime, demanding higher performance at a lower cost. Recent statement in a bulletin article published by European commission reads: "... When Europe goes broadband mobile, efficient radio access technologies are a must" [1].

In addition to this growing demand on radio resources, some new technical challenges are also emerging. For instance, conventional cellular networks are single-hop based i.e. normally only "direct transmission" is employed for the radio connections between terminals and base stations. It is widely anticipated that future cellular systems will deploy multi-hop concept [2] which in turn will increase the complexity of Radio Resource Management (RRM).

In order to satisfy this growing demand and complexity, new RRM methods providing more efficient ways of using spectral resources are required. In other words, RRM in B3G cellular systems has to provide means for optimal usage of the allocated radio spectrum.

Packet scheduling, Admission control, Congestion control (and Routing in case of multi-hop) are some of those key RRM functionalities that need to be well adapted to this growing demand on the radio spectrum and provide more efficient resource utilization.

The objective in this PhD thesis is to investigate and develop novel uplink packet scheduling algorithms for future cellular systems in order to achieve efficient resource utilization. Several
existing methods for packet scheduling in single-hop “Enhanced Uplink UTRA-FDD” [3] are presented, evaluated and compared. Enhanced Uplink UTRA-FDD (also called HSUPA) is chosen here only as a research platform representing an interference limited system but the results are applicable to other systems as well.

Uplink scheduling has been a very active research topic in B3G systems such as HSUPA. In HSUPA, scheduling consists of several functionalities. A number of techniques have been proposed for uplink scheduling, including centralised scheduling performed in Radio Network Controller (RNC) or decentralised scheduling performed in NodeB.

In earlier 3GPP releases Rel99 [4], Rel4 [5] and Rel5 [6], uplink scheduling and rate control resides in RNC. Hence all users would be scheduled at RNC level and this method is called Centralized scheduling. An example of a Centralized scheduling is presented in [7]. It is also possible for base station (NodeB) to control user’s transmission rate/time. By providing the NodeB with appropriate measures, tighter control of the uplink interference is possible which in turn, may result in increased capacity and improved coverage. This method is called Decentralized scheduling since all users would be scheduled by their server NodeB, and not by RNC. Recent results in [3] show better performance for decentralized scheduling compared with centralized one especially in terms of delay performance.

Several decentralized scheduling algorithms are studied in this thesis and their performance are evaluated and compared. In addition, a combined algorithm called Virtually Centralized Scheduling (VCUP) algorithm is presented and investigated.

Extensive rounds of simulations have been carried out and simulation results provided in order to evaluate the performance of these scheduling techniques and to compare them.

In an interference limited CDMA-based system, the uplink cell capacity is basically limited to the received uplink interference, usually called Rise over Thermal noise (RoT). State of art NodeB scheduling algorithms for Enhanced Uplink UTRA-FDD system suggest that, each NodeB assigns radio resources (rate and time) to its users on a priority basis until the estimated RoT reaches its pre-defined target. The main advantage of decentralized scheduling is to reduce the level of overhead signalling required in centralized scheduling and hence increase spectral efficiency. In this thesis, we extend this advantage further by developing Distributed scheduling algorithms in which User Equipment (UE) can decide on its uplink transmission rate instead of RNC or NodeB.

It will be shown that the main shortcoming in decentralized and/or distributed scheduling is that a considerable proportion of RoT comes from inter-cell interference which NodeB has little knowledge about and control upon. We provide a detailed look into this problem and highlight the importance of intercell interference control as the key factor in future radio resource management.
The summit of this thesis is to address this problem by introducing a novel technique called Load Matrix. An innovative technique for resource scheduling is presented using Load Matrix (LM) that enables efficient resource utilization. In its simplest form of deployment, a central RRM entity employs this matrix for radio resource allocation in conjunction with interference management purposes. We then compare the network performance of decentralized packet scheduling with Load Matrix technique.

Load Matrix database consists of corresponding load factors of each active User Equipment (UE) in each and every cell in the network. It can be used for all interference limited systems but as stated earlier, here the focus is on enhanced uplink UTRA-FDD to demonstrate the concept and evaluate the performance. System level simulation results presented show that this technique could ensure overwhelmingly better performance in terms of interference outage and cell throughput.

The Load Matrix and its evaluation results are first investigated based on perfect knowledge of channel information from all the active users without additional delay in order to explore the upper-bounds of system capacity in this technique. We then study the effects of data impairments in the Load Matrix database and its consequences on the system performance namely cell throughput, Interference outage and delay. It is interesting from practicality point of view to also look at implementation issues such as signalling overhead, sensitivity of Load Matrix to channel measurement errors and signalling delay. It will be shown that Load Matrix database is in fact a very sparse matrix and therefore the volume of LM data elements required for this technique and eventually the signalling overhead is actually small. Another practical impairment studied is channel prediction and/or measurement error. The effect of channel measurement and prediction error is studied and the system performance is compared in different error levels. Given the fact that LM database is a very sparse matrix, we then study the effect of LM scheduling while being restricted to neighbouring cells only and extend the results of channel error to this study as well. Finally the effect of additional delay in the channel information used by LM is studied and simulation results are discussed.

1.2 Contributions and Achievements

1.2.1 Two new distributed scheduling algorithms

In this thesis, we extend the advantage of fast scheduling by developing two Distributed scheduling algorithms in which User Equipment (UE) decides its uplink transmission rate instead of RNC or NodeB. Algo-1 uses two basic metrics i.e. Comparative Buffer (CB) and Capacity Indicator (CI). CB is user-specific based on comparative UE’s buffer size, and CI is an indicator
for uplink capacity at Node-B and hence is cell-specific. In Algo-2, CI metric is still used but CB is replaced with normalized UE queue size which is locally available at UE and therefore eliminating the additional signalling overhead. As the result, overhead signalling in Algo-2 is minimized due to the fact that CI is a cell-specific value and can be sent over broadcast channel with minimal signalling overhead. Simulation results provided showing distributed scheduling technique introduced in this thesis can achieve superior performance (30% increased cell throughput on average) compared against conventional algorithms. It is observed when the system load increases, there is little advantage compared to traditional single-cell scheduling algorithms.

1.2.2 Identified the shortcomings of traditional scheduling (single-cell based)

It is shown in details that there is an inherent shortcoming in any decentralized/distributed single-cell scheduling technique: A considerable proportion of RoT comes from inter-cell interference which NodeB has little knowledge about and control upon. We provide a detailed look into this problem and highlight the importance of Intercell interference control as the key factor in future radio resource management.

1.2.3 Introduced Load Matrix technique (multi-cell based)

We then address the influence of multi-cell interference on overall radio resource utilisation and propose an innovative novel technique to uplink scheduling, setting a new direction for future research on resource scheduling strategies in multi-cell scenarios. A technique called Load Matrix (LM) is proposed which facilitates joint management of interference and allocation of radio resources within and between cells. Simulation results show significant improvement in the resource utilization and overall network performance. Using the LM technique, average cell throughput can be increased as much as 30% compared to a conventional benchmark algorithm. Results also show that maintaining cell interference within a margin instead of a hard target can significantly improve resource utilization.

LM has been awarded by Nokia in the Research Excellence Awards Competition in CCSR, University of Surrey, 2006.

1.2.4 LM full analysis under ideal and practical assumptions

Load Matrix performance is examined in a variety of practical conditions as opposed to ideal condition. It was shown that Load Matrix is in fact a sparse matrix in nature and therefore the number of load elements which are significant enough to be considered for this technique and eventually the signalling overhead is reasonably small. In other words, LM scheduling can be performed satisfactorily by coordination amongst adjacent cells only, when reliable channel
estimation/measurement techniques with error margin less than 2 dB STD is used. Overall, by comparing LM performances with and without impairments studied in this thesis, it is shown that LM scheduling performance in terms of interference outage, throughput and packet delay is resilient to channel estimation/measurement error of up to 5 dB standard deviation. In addition, it is proved that LM scheduling performance in terms of interference outage, throughput and packet delay is resilient to channel information delay of up to 2 Transmission Time Interval (TTI).

1.2.5 Stability of system performance achieved by LM technique

It is proved that both cell interference and cell throughput can and will be stabilized as direct result of multi-cell interference control by implementing Load Matrix. It is shown when LM scheduler is implemented, basically there is no "best" and "worst" cell like conventional systems as LM manages interference level very close to its target for all cells at all time. Comparing benchmark cases with LM, it is shown that Load Matrix has not only increased cell throughput (as much as 50% for some cells) but also stabilizes that. Load Matrix provides uniformity and longevity in interference outage and throughput performance, consistently maintains it within a narrow range of variation both in time and across the whole network.

Although HSUPA system level simulator was used in producing the performance results, Load Matrix is a generic concept adaptable to other interference-limited systems.

1.3 Thesis Outline

This thesis is organised as follows: An overview on packet scheduling in uplink (HSUPA system) is presented in chapter 2. Several decentralized scheduling algorithms are studied and their performance are evaluated and compared. In addition, a combined algorithm called Virtually Centralized Scheduling (VCUP) is presented and investigated. Two referenced decentralized scheduling algorithms proposed by Nokia and Qualcomm are studied and their performance is evaluated. Section 2.2 provides detailed description of Nokia's Node B scheduling algorithm. Qualcomm's Node B scheduling algorithm which has been used in [3] as reference algorithm, is presented in section 2.3. Finally, a combined algorithm proposed by Fujitsu (called VCUP) is presented in section 2.4.

In chapter 3, we try to extend the advantage of distributed scheduling further by developing novel Distributed scheduling algorithms in which User Equipment (UE) decides on the uplink transmission rate instead of RNC or NodeB. Section 3.2 and 3.3 present two examples of these distributed scheduling algorithms.

Extensive rounds of simulations have been carried out, in order to evaluate the performance of these scheduling techniques and to compare them. Chapter 4 provides a comprehensive look into
system level performance of conventional schedulers and distributed scheduling algorithms previously presented, through variety of simulation results under different load conditions. Selected set of results provided in chapter 4 and complete set of results in appendix A.

The importance of intercell interference is investigated in Chapter 5. It is shown that the main shortcoming in decentralized/distributed scheduling technique is that a considerable proportion of RoT comes from inter-cell interference which NodeB has little knowledge about and control upon. In this chapter, we provide a detailed look into this problem and highlight the importance of Intercell interference control as they key factor in future radio resource management. We discuss details of intercell interference problem and its severe impact on scheduling in chapter 5, and then address this problem by introducing an effective resource allocation technique called Load Matrix (LM).

The summit of this thesis, Load Matrix, is presented in chapter 6. LM is an innovative technique towards resource scheduling that enables joint interference control and efficient resource utilization. In its simplest deployment, a central entity employs this matrix for radio resource allocation in conjunction with interference management purposes. We then compare the network performance of decentralized packet scheduling with Load Matrix technique. It will be proved in chapter 6 that finding the optimum uplink resource allocation is in fact a Non deterministic Polynomial time (NP)-hard problem in both single-cell and multi-cell cases. Load matrix novel technique is then provided as a practical solution to efficient resource allocation problem.

Basically, Load Matrix database consists of corresponding load factors of each active user in every cell within the network. Section 6.1 presents LM technique, section 6.2 defines LM details and section 6.3 provides System level simulation results showing that this technique could ensure better performance in terms of interference outage and cell throughput. Section 6.4 compares performance with and without Load Matrix. Extensive simulation results on interference outage, throughput and packet delay performance of a reference decentralized scheduling [8] used in [3] together with the proposed Load Matrix technique are provided and compared.

The Load Matrix and its evaluation results discussed in chapter 6 are based on perfect knowledge of channel information from all the active users without additional delay. In chapter 7, we study the effects of data impairments in the Load Matrix and its consequences on the system performance namely cell throughput, Interference outage and delay. It should be noted that LM presentation in previous chapters was based on perfect knowledge of channel information and without considering additional delay in order to explore the upper-bound limits of system capacity in this technique. It is interesting however from practicality point of view to further study the implementation issues such as signalling overhead, sensitivity of Load Matrix to channel measurement errors and signalling delay. In this chapter, first a closer look at Load Matrix is
provided and load element projection is discussed. It shows that Load Matrix is in fact a very sparse matrix and therefore the volume of required LM elements for this technique and eventually the signalling overhead is small. Another practical impairment studied in chapter 7 is the effect of channel prediction and/or measurement error in LM scheduling. The effect of channel measurement and prediction errors is studied and system performance is compared in different error conditions. Given the fact that LM is a very sparse matrix, we then study the effect of LM scheduling while LM database is restricted to neighboring cells only and extend the results of channel error to this study as well. Finally the effect of additional delay in the channel information used by LM is studied and simulation results are discussed.

Another important feature of LM scheduling is its Stability over time which is presented and explored in Chapter 8. In this chapter we show the consistency of Load Matrix performance over time (longevity) and across the network (uniformity) in terms of throughput and RoT and also compare it with benchmark scheduling algorithm. Finally chapter 9 provides the conclusion of the thesis.

1.4 Summary

In this chapter, the motivation and objectives as well as novel contributions and structure of the thesis are provided. It is explained that in order to satisfy the growing demand for higher bandwidth needed by emerging new applications as well as complex QoS requirements, new RRM methods providing more efficient ways of using spectral resources are required. In other words, RRM in future cellular systems has to provide means for optimal usage of the allocated radio spectrum. Packet scheduling, Admission and Congestion control are some of those key RRM functionalities that need to be well adapted to this growing demand on the radio spectrum and provide more efficient resource utilization.

The objective in this thesis is to investigate and develop novel uplink packet scheduling algorithms for future cellular systems in order to achieve efficient resource utilization. Main contributions are two folds: first, in single-cell scenario, new distributed scheduling introduced and novel algorithms developed and assessed. Second, we address the influence of multi-cell interference on overall radio resource utilisation and propose an innovative novel technique to uplink scheduling, setting a new direction for future research on resource scheduling strategies in multi-cell scenarios. A technique called Load Matrix (LM) is proposed which facilitates joint management of interference and allocation of radio resources within and between cells.

Chapter 2 provides in-depth overview on packet scheduling in uplink applied in HSUPA system. In chapter 3, we extend the advantage of decentralized scheduling further by developing novel distributed scheduling algorithms in which each user decides on the uplink transmission rate.
instead of RNC or NodeB. Chapter 4 provides a comprehensive look into system level performance of conventional schedulers and distributed scheduling algorithms previously presented, through variety of simulation results under different load conditions. Details of intercell interference in Multi-cell scenario is investigated in Chapter 5, and then addressed by introducing a novel effective resource allocation technique called Load Matrix (LM) in chapter 6. In chapter 7, we study the effects of practical impairments in the Load Matrix and its consequences on the system performance namely cell throughput, Interference outage and delay. Another important feature of LM scheduling is its Stability over time which is presented and explored in Chapter 8. Finally, chapter 9 provides the conclusion of the thesis together with suggested directions for future research work.
Chapter 2

2 Scheduling in Uplink

2.1 Introduction

Mobile cellular systems are facing new challenges created by the demand for emerging services and applications. Wide range of services with diverse Quality of Service (QoS) requirements is becoming more popular and widely used. The demand for higher bandwidth and data rates has been increased substantially during recent years. This has made it vital for future mobile cellular systems to implement efficient resource allocation techniques.

In order to achieve efficient resource utilization in all sorts of deployment scenarios and QoS requirements in the future wireless cellular systems, new resource allocation methods must be developed. In other words, resource allocation has to deliver close to optimum utilization of the available radio spectrum in the next generation of cellular wireless systems regardless of deployment scenarios and conditions.

Importance of resource scheduling was appreciated with the support of high data rate services in the evolution of UMTS standard [4] to High Speed Downlink Packet Access (HSDPA) [9] and Enhanced Uplink [3].

A variety of resource allocation strategies and techniques, mainly for downlink, can be found in references [10]-[15]. In [10] a system with multiple traffic classes was considered and resource allocations were based on the specific characteristics of traffic flows resulting in minimization of power consumption or maximization of system capacity. Under mixed service traffic including both real-time and non-real time services, efficient resource allocation from a shared resource pool is a challenging task due to varied and stringent QoS requirements. In [11] authors proposed a fixed resource partitioning method in which total resource pool was partitioned between different service classes and independent resource schedulers were responsible for each resource partition whereas in [12], scheduling was more unified and partitioning was dynamic to enhance spectral efficiency.
Another technique towards resource allocation, called utility based technique, tries to maximize the total network utility and thereby enhancing resource allocation. For example, pricing is a well-known utility function used in [13] for resource allocation. In [14] authors used user's QoS as utility function and then convert the resource allocation problem into a non-cooperative game where each user tries to maximize its own utility. A downlink resource allocation method based on dynamic pricing was proposed in [15] aiming to maximize the summation of users' utility.

On the link level, adaptive transmission is one of the most recent technologies being investigated for enhancing the spectral efficiency in future cellular systems [16]. Fast scheduling together with adaptive modulation-coding, facilitates exploitation of channel variations resulting in multi-user diversity gains [17]. This technique takes advantage of instantaneous channel conditions of different users where the channel fading are relatively independent. Adaptive transmissions are more effective for low mobility users compared with fast moving users' channel.

In this thesis, our focus is solely on uplink. Uplink resource allocation methods can be categorized as centralized or decentralized in terms of the network location/node in which scheduling takes place. In Universal Mobile Telecommunications System (UMTS) for example, if the scheduler resides in RNC, it is called centralized and if it resides in base station it is called decentralized.

In an interference-limited system such as UMTS, the uplink cell capacity is basically limited by the total received uplink power at the base station due to the transmit power limitation of user terminals [18]. In decentralized scheduling, each base station assigns radio resources to its users on a priority basis until the estimated Rise over Thermal noise (RoT) level reaches a pre-defined target. Recent studies in Enhanced Uplink UTRA, also called High Speed Uplink Packet Access (HSUPA), show that decentralized scheduling has better performance compared with centralized one [3]. The subject of centralized versus decentralized scheduling has been studied extensively in recent years both in 3rd Generation Project Partnership (3GPP) standard body for HSUPA and in the literature [3][19]. In [20] the performance of centralized packet scheduler of the UMTS system is evaluated while in [19] the performance of a decentralized scheduling is evaluated and compared with the centralized one in [20].

The basic advantage of decentralized over centralized technique is due to its fast response to dynamic and fast varying nature of mobile environment. However, the decentralized scheduling algorithms have an inherent shortcoming due to their vulnerability to intercell interference, which has not been addressed adequately yet. Considerable proportion of RoT at the base station is made up from multiple access intercell interference which the base station has little knowledge about or control upon. This in turn may lead the system to interference outage and poor resource utilization particularly when interfering cells have similar pattern of traffic load variations.
Inadequate (intercell) interference management, particularly in highly loaded systems, is an inherent problem of decentralized scheduling regardless of the algorithm being used. Several interference mitigation techniques such as Multi User Detection (MUD) [21], Interference cancellation (IC) [22], antenna beamforming [23], and their combinations have been studied extensively and proved to be effective in mitigating interference to some extent and thereby increase system capacity. However, in a highly loaded system, the problem of intercell interference remains an important issue. For instance, MUD with Minimum Mean Square Error (MMSE) detection MMSE-MUD is recognised as an effective interference suppression technique for increasing the system capacity [24]. It has been demonstrated in [25] that MMSE-MUD can achieve good performance in single-cell scenario.

From the scheduling perspective, although intercell interference problem is more severe in decentralized scheduling, it is also present in conventional centralized scheduling due to the fact that the intercell interference impact of a scheduled user is not known and therefore has not been considered by the central scheduler.

In this chapter, we review some of the well known uplink scheduling techniques used as a reference in the literature and 3GPP standards. We also present an example of newly emerged semi-distributed scheduling algorithms.

In earlier 3GPP releases Rel99 [4], Rel4 [5] and Rel5 [6], the uplink scheduling and rate control resides in RNC. Since all users would be scheduled in RNC and centrally controlled transmission rate would be assigned to them, the method is usually referred to as Centralized scheduling. An example of a Centralized scheduling is presented in [7].

In contrast to centralized scheduling, it is also possible for the NodeB (base station) to control scheduling and rate assignment for its own user terminals. This method is therefore called decentralized scheduling. First, RNC determines the full set of transmission rates in the form of Transport Format Combination Set (TFCS). TFCS is a set formed by a number of elements called Transport Format Combination (TFC) each representing a unique transmission rate. RNC then sends a pointer to each Node B indicating its maximum allowed TFC in the TFCS. This basically limits the maximum transmission rate that Node B can assign to its users. Next, Node B assigns suitable transmission rate to each and every individual user in its cell, within the limits set by the RNC pointer, taking into account UE’s data buffer occupancy and its available transmit power.

Since all users would be scheduled according to their server NodeB and not by RNC, this method is decentralized. Numerous investigations and simulations have been carried out to compare the performance of these methods. For instance, recent results in [3] show better performance for decentralized scheduling compared with centralized one especially in terms of delay performance. So, to start, it is worth to look into some of well known decentralized scheduling algorithms.
Chapter 2. Scheduling in Uplink

Figure 2-1 and Figure 2-2 illustrate the TFC assignment in HSUPA centralized and decentralized scheduling respectively.

Decentralized scheduling algorithms can all be viewed as management of the TFC selection in the UE and mainly differs in how the Node B can influence this process and the associated signalling requirements. The set of TFCs from which the UE may choose a suitable TFC is called Node B controlled TFC subset. It should be noted that Node B controlled TFC subset is restricted to the
boundaries set by RNC meaning the maximum rate is limited by the Node B pointer set by RNC (see Figure 2-2). Each TFC in TFCS represents one transmission rate. In Figure 2-3, the TFCs in a TFCS are shown in descending order with respect to the transmission power required. The maximum rate in the Node B controlled TFC subset is called UE pointer. Any TFC equal to or below the UE pointer can be selected by the UE, provided there is sufficient power headroom at UE and sufficient data available in the UE's buffer. The Node B itself is restricted by Node B pointer, which is assigned by RNC.

![Diagram of TFC pointers](image)

Figure 2-3: Definition of the TFC pointers

In Figure 2-3, the TFC0 represents the highest transmission rate and TFC10 the lowest. Required transmission power depends on the target SINR at the receiver (base station) on the given data rate and the target BLock Error Rate (BLER). These target SINRs can be obtained from the BLER vs. SINR performance curves in the physical layer (also known as link-to-system performance curves [3]). Based on speech quality measurements, target BLER is usually set to 1% in order to provide acceptable QoS for voice communications. However, it is shown in [26] that QoS in voice communication is highly perceptual. Depending on the content of speech, sometimes BLER of up to 10% can still provide acceptable quality in voice communication. This is an interesting subject leading to overall system capacity enhancement from a different angle but out of scope in this thesis.

The transmission rate in the uplink is the result of TFC selection algorithm in UE. TFC selection is a Medium Access Control (MAC) function that UE uses to select a TFC from its Node B controlled TFC subset whenever it has something to transmit. TFC is selected based on the need for data rate i.e. UE's buffer contents, currently available transmission power headroom, available TFCS and the UE's capabilities. The details of TFC selection function in HSUPA remains same as release99, see [4]. It is worth however, to briefly look at the timing of TFC selection (rate assignment) process in HSUPA system especially to understand how frequent this selection may occur. Figure 2-4 illustrates an example of a timing diagram for UE request, scheduling and TFC assignment in a decentralized scheduling. As mentioned earlier, TFC selection is a MAC
functionality. In MAC, as a sub-layer of Data Link layer, the time unit for an independently decodable transmission data block is called Transmission Time Interval (TTI). TTI is a basic time unit often used in MAC functions including TFC selection and scheduling. In Rel99[4], the shortest TTI is 10 ms where as in HSPA [3], TTI of 2 ms is also considered. Scheduling period is always a multiple integer of TTI, therefore it is appropriate to use TTI as our basic time unit. In centralized scheduling due to delay restrictions, scheduling period can be tens of TTIs. For instance, in [7] scheduling period of 200ms equal to 20 TTI is suggested. In decentralized scheduling however, since NodeB controls the process and delay is far less, the scheduling can be updated every TTI as shown in Figure 2-4.

Figure 2-4: Timing diagram for request, scheduling and assignment in decentralized scheduling

(Scheduling period = 1 TTI)

In the rest of this chapter, two main decentralized scheduling algorithms proposed by Nokia [20] [27] and Qualcomm [8] are studied. Section 2.2 provides detailed description of Nokia’s Node B scheduling algorithm. Qualcomm’s Node B scheduling algorithm which has been used in [1] as reference algorithm, is presented in section 2.3. Finally, a combined algorithm by Fujitsu called Virtually Centralized Scheduling (VCUP) algorithm [28][29][30] is presented in section 2.4. As part of investigation undertaken, extensive rounds of simulations have been carried out in order to evaluate the performance of these scheduling techniques and to compare them. Detailed simulation results for Nokia’s Node B scheduler, Qualcomm Node B scheduler and VCUP are presented and their performances are compared in chapter 4 and Appendix A.
2.2 Nokia’s NodeB Scheduling Algorithm

2.2.1 Model Structure

The Nokia’s Node B scheduling algorithm is based on a factor derived at the Node B called Resource Utilisation Factor (RUF) [20][27]. RUF is a counter bringing information on the number of last consecutive transmissions with either selected transmission rate equal to zero (negative value) or selected data rate corresponding to the maximum allocated data rate at Node B (positive value). The RUF counter is reset to zero if the selected transmission rate by UE is both different from zero and from the maximum allocated rate at the Node B (i.e. NodeB pointer). The RUF is updated at Node B on TTI basis.

Node B Scheduler categorizes the active UEs into three groups: DOWN, UP and KEEP. The decision rules for the considered Node B scheduling algorithm are defined along these lines:

**DOWN** — In order to reduce the transmission rate allocated to inactive users, a threshold value $T_{down}$ is defined. UE’s transmission rate would be downgraded to the minimum when RUF is less than or equal to $-T_{down}$. Moreover, in order not to allocate data rates higher than the one that UE can support, UEs with $-T_{down} < RUF < 0$ are downgraded using a downgrading step of 1.

**UP** — A UE is selected as a candidate for data rate upgrade if $RUF > T_{up}$. In the upgrading procedure, UEs are progressively upgraded up to the maximum possible transmission rate, under the constraint of available power and fairness among users. Higher priority is given to those UEs that are allocated lower data rates (i.e. Fair Resource Scheduling).

**KEEP** — No actions are undertaken for this class of UEs.

$DOWN$: candidates for transmission rate downgrade to minimum

$DOWN$: candidates for transmission rate downgrade with 1 step

$UP$: candidates for transmission rate upgrade

$KEEP$: no action

Figure 2-5 illustrates the decision regions for the RUF counter.
The task sequence performed by the Node B is as follows: First the downgrade actions are carried out. Then the candidates for data rate increase are upgraded. In the upgrading procedure, a power increase estimator is used to estimate the noise rise (NR) after each data rate allocation. If the estimated NR exceeds a predefined NR target (NRT), the data rate upgrade is not allowed. NR is defined as the ratio of the total received wideband power to the noise power.

In the case of Soft Hand Over (SHO), UEs in soft handover are assumed to independently receive data rate allocations from all the Node Bs in the active set. Each UE then selects the one which is allocated by the best server Node B.

### 2.2.2 Algorithm Description

At TTI instant $T$, the RUF counter of each UE i.e. $RUF(T)$ is computed as follows:

- **if** (Selected Transmission Rate $= 0$ kbps)
  - if $(RUF(T-1) < = 0)$ then $RUF(T) = RUF(T-1) -1$
  - else $RUF(T) = -1$
- else **if** (Selected Transmission Rate $= \text{Maximum Allocated Data Rate at Node B}$)
  - if $(RUF(T-1) > = 0)$ then $RUF(T) = RUF(T-1) +1$
  - else $RUF(T) = 1$
- else $RUF(T) = 0$

Associated with each UE is a priority counter. Priority of each UE is initialized to 0 in the beginning. At each scheduling instant, for each cell, first the rate of all DEs in the DOWN group would be decremented according to their RUF value. Then the cell capacity is updated to be used by UP group users. UP group users would be sorted based on a priority function.

The scheduling algorithm steps are as follows:

1. Categorize all the UEs in each cell based on their RUF values.
2. Assign decremented rate to DOWN group UEs based on its RUF value.
3. Update the available cell capacity for the released resources by DOWN group UEs.
4. Prioritize the UP group UEs. UEs that have been allocated lower data rates are served first.
5. Set $k=1$.
6. The UE at the $k^{th}$ position in the priority list is assigned the new rate if the noise rise does not exceed the NRT.
7. Update the available capacity.
8. $k = k+1$; if $k < \text{total number of UEs in the priority list}$, Go to Step 6, otherwise, stop.

In Chapter 3, we will provide the performance results for Nokia's NodeB scheduler alongside other scheduler algorithms presented in this chapter for comparison.
Chapter 2. Scheduling in Uplink

2.3 Qualcomm’s NodeB Scheduling Algorithm

2.3.1 Model Structure

The Qualcomm Node B scheduler [8], located in NodeB, maintains a list of all UEs that are served by or in soft hand over with that NodeB. Scheduler assigns resources only to the UEs for which the NodeB has the best downlink (i.e. best server NodeB).

The following is the summary of the scheduling procedure:

- Update queue information for each UE it schedules
- Compute the maximum TFC allowed in TFCS for each UE it schedules
- Update the available resources
- Make a priority list according to the predefined priority function
- Perform greedy filling for maximum capacity utilization. The right to transmit on the uplink is granted to the highest priority UE first, then successively to lower priority UEs.

Associated with each UE, the scheduler stores an estimate of UE’s queue size $\hat{Q}$ and maximum scheduled rate $R_{\text{max}}(s)$ at every scheduling instant $s$.

Scheduler estimates the maximum TFC allowed in TFCS for the UE as follows [8]:

$$\hat{Q}(f) = \hat{Q} - (R_{\text{assigned}}) \cdot \left(\frac{\text{ActionTimeDelay}}{TTI}\right) \cdot TTI[\text{ms}] \tag{2.1}$$

$$R_{\text{max}}(s) = \min \left\{ R_{\text{max}}(\text{power}), \arg \max_{R \leq \text{304Kbps}} \left\{ \left( EDPDCH \_PRD / TTI[\text{ms}] \right) \right\} \right\} \tag{2.2}$$

where $R_{\text{assigned}}$ is the maximum transmission rate allowed in TFCS during the current scheduling period and UE is allowed to transmit until the ActionTime of the next assignment. ActionTime is the time delay between UE sending the request to NodeB and receiving the new allocation rate for transmission. $\hat{Q}(f)$ is the estimated data queue size for the UE, $EDPDCH\_PRD$ is the scheduling period (in which UE is allowed to transmit on the E-DPDCH channel [3]), and
\[ R_{\text{max}} (\text{power}) \] is the maximum rate allowed in TFCS that the UE can support given its power headroom only, regardless of its buffer size.

In order to guarantee throughput-wise fairness among users, the priority function is defined in terms of the proportional fairness [31]. For \( k \)th UE:

\[
PRIORITY_k = \frac{R_{\text{max}}}{r_k}
\]

(2.3)

where \( r_k \) is the average transmission rate already allocated to the UE \( k \) by the scheduler, and \( R_{\text{max}} \) is the maximum rate allowed in TFCS based on UE's buffer size and power constraints. \( r_k \) is the average allocated transmission rate and it is updated at every scheduling interval as:

\[
r_k(t+1) = (1 - \frac{1}{T_c})r_k(t) + \frac{1}{T_c} R_k
\]

(2.4)

where \( T_c \) is a time constant set to 10 [8]. \( R_k \) is the current allocated maximum TFC allowed in TFCS of the \( k \)th UE.

In the case of SHO, UEs in soft handover are assumed to independently receive data rate allocations from all the NodeBs in their active set. Each UE then selects the one which is allocated by its best server Node B.

2.3.2 Algorithm Description

As mentioned before, associated with each UE is a priority counter \( PRIORITY \). \( PRIORITY \) of a user is initialized to 0 in the beginning of transmission. When a new UE (e.g. UE\(_m\)) enters the system with cell\(_i\) as its best server, its \( PRIORITY_m \) is set to:

\[
PRIORITY_m = \min \{PRIORITY_i, \forall i \text{ such that UE}_i \text{ has cell}_j \text{ as the best server}\}
\]

(2.5)

meaning the UE will have the lowest priority for transmission amongst all active users already exist in cell\(_j\). The scheduling algorithm steps are as follows:

1. Set \( k=1 \).
2. The UE at the \( k \)th position in the priority list is assigned the rate \( R_k \) given by:
Chapter 2. Scheduling in Uplink

$$R_k = \min \left\{ R_{\text{max}}(s), \arg \max_j \begin{bmatrix} R \left| C_{\text{module}}(j) \right. \cdot \frac{\text{Sinr}(R)}{1 + \text{Sinr}(R)} + \frac{\text{Sinr}(\max(0, R_{\text{auto}}))}{1 + \text{Sinr}(\max(0, R_{\text{auto}}))}, \quad j \text{ is the scheduling sector} \end{bmatrix} \right\} \quad (2.6)$$

3. The available capacity is updated to:

$$C_{\text{module}}(j) = C_{\text{module}}(j) - \frac{\text{Sinr}(R_k)}{1 + \text{Sinr}(R_k)} + \frac{\text{Sinr}(\max(0, R_{\text{auto}}))}{1 + \text{Sinr}(\max(0, R_{\text{auto}}))}; \quad (2.7)$$

$$j \text{ is the scheduling sector}$$

4. Calculate new PRIORITY$_k$ as explained in (2.3).

5. $k = k + 1$; if $k <$ total number of UEs in the list, Go to Step 2.

In Chapter 4 and appendix A, the performance results for Qualcomm’s NodeB scheduler is provided alongside other scheduler algorithms presented in this chapter for comparison.

2.4 Virtually Centralised Scheduler (VCUP)

2.4.1 Model structure

VCUP [28][29][30] is an interesting scheduling algorithm as it is not a fully decentralized scheduling algorithm. It is better to call it a semi-distributed scheduling since VCUP is basically a fine tuning algorithm for TFC selection in UE, combined with performing decentralized scheduling at Node B. In VCUP, NodeB sends the TFCS subset to all its UEs rather than UE pointer. After receiving the TFCS subset from the Node B scheduler, each UE will perform TFC selection to choose the appropriate transmission rate according to its buffer size and available power. VCUP is a simple mechanism in addition to TFC selection at UE which tries to take into account the competition in the cell by means of a Comparative Metric (CM). In order to achieve better QoS and fairer scheduling decision, Node B creates CM values for each UE using, for example, a combination of the buffer status information received from UEs. This metric shows how much congestion is faced by each UE in uplink. Node B then sends CM alongside TFCS to each UE.
In VCUP algorithm, Node B calculates CM for all the UEs in its cell and sends it back to them. At the UE, received CM will be mapped to a TFC within TFCS subset after TFC selection. That means UE first employs its own transmit power status and buffer to determine the maximum available rate (maximum allowed TFC), within the restricted TFC subset, according to TFC selection shown in Figure 2-6.

![TFCS Subset sent by NodeB](image1)

**Figure 2-6: First stage of rate selection (same as in TFC selection Release 99)**

UE then performs the second stage of rate selection which is to come up with the final TFC as it is shown in Figure 2-7. It maps the CM value and then uses a predefined set of $\alpha$ values to determine the final TFC. UE chooses its uplink transmission rate by comparing CM with $\alpha$ values.

![UE Allowed TFC Subset](image2)

**Figure 2-7: Second stage of rate selection in VCUP (using CM)**
2.4.2 Algorithm Description

At every TTI, UE determines the amount of data existing in its buffer and waiting for transmission. Assuming that the $n^{th}$ UE belongs to the $j^{th}$ group of service with the same delay tolerance and the same maximum buffer size, UE divides this value by the maximum length of its data buffer:

$$Norm_{_B}uf_{n,j}(m) = \frac{Bu_{n,j}(m)}{Bu_{max,j}} , \quad n=1...N_j$$  \hspace{1cm} (2.8)

where index $m$ represents current TTI or uplink scheduling event, $N$ is the total number of the UEs, $Bu_{max,j}$ is the maximum packet data buffer length which depends on the service group $j$.

UE multiplies this value $Norm_{_B}uf_{n,j}(m)$ by 100, takes the integer part and sends the result to Node B, which represents a ratio between zero and one.

Node B first classifies all UEs based on the application or class of service. Then it determines the distance from minimum normalized buffer size:

$$Dist_{min_B}uf_{n,j}(m) = Norm_{_B}uf_{n,j}(m) - Norm_{min_B}uf_{n,j}(m) , \quad n=1...N_j$$  \hspace{1cm} (2.9)

Node B then normalizes this value to the maximum distance:

$$Dist_{min_B}uf_{n,j,norm}(m) = \frac{Dist_{min_B}uf_{n,j}(m)}{Dist_{min_B}uf_{max,j}(m)} , \quad n=1...N_j$$  \hspace{1cm} (2.10)

In a similar way, distance from average occupancy is determined in Node B:

$$Dist_{avg_B}uf_{n,j}(m) = Norm_{_B}uf_{n,j}(m) - Norm_{avg_B}uf_{avg,j}(m) , \quad n=1...N_j$$  \hspace{1cm} (2.11)

Since the value of $Dist_{avg_B}uf_{n,j}(m)$ might be negative, a positive bias value is added to produce a positive value between 0 and 1:

$$Dist_{avg_B}uf_{n,j}(m) = Dist_{avg_B}uf_{n,j}(m) + 1.0 , \quad n=1...N_j$$  \hspace{1cm} (2.12)

Then:

$$Dist_{avg_B}uf_{n,j,norm}(m) = \frac{Dist_{avg_B}uf_{n,j}(m)}{Dist_{avg_B}uf_{max,j}(m)} , \quad n=1...N_j$$  \hspace{1cm} (2.13)

$Dist_{min_B}uf_{n,j,norm}(m)$ and $Dist_{avg_B}uf_{n,j,norm}(m)$ can be represented as Comparative Metrics (CM): $CM_{n,j} = Dist_{min_B}uf_{n,j,norm}(m)$ and $CM_{n,j} = Dist_{avg_B}uf_{n,j,norm}(m)$.
Chapter 2. Scheduling in Uplink

The values $CM_{n1,j}$ and $CM_{n2,j}$ are then signalled from Node B to UEs either independently or in a combined fashion. The combination in NodeB reduces the amount of signalling. In this case, only one CM value is signalled from NodeB to each UE.

Under the proposed combination, Node B first determines:

$$C_{n,j}(m) = CM_{n1,j} \cdot CM_{n2,j}, \quad n = 1 \ldots N_j$$  \hspace{1cm} (2.14)

Node B then normalizes the result to the maximum combined value:

$$CM_{n,j}(m) = C_{n,j} / C_{max,j}, \quad n = 1 \ldots N_j$$  \hspace{1cm} (2.15)

In the case of Soft Hand Over (SHO), UEs in soft handover are assumed to independently receive CM from all the Node Bs in their active set. Each UE then selects the one which is allocated by the best server NodeB.

The $\alpha$ values in Figure 2-7 are between 0 and 1 with a step defined as:

$$\Delta \alpha = 1/u$$  \hspace{1cm} (2.16)

For example, if $u$ or the maximum number of available TFCs in the TFC subset is 5 then the $\alpha$ values are:

$$\alpha_2 = 0.2, \alpha_4 = 0.4, \alpha_3 = 0.6, \alpha_2 = 0.8, \alpha_1 = 1.0$$  \hspace{1cm} (2.17)

UE compares its CM value to $\alpha$ values and maps it to the closest one to choose its uplink transmission rate.

The CM calculation presented above is an example for a comparative metric based on UE buffer occupancy (we refer to it as CM2). One can think of other possibilities to incorporate the competition element in VCUP algorithm by considering other parameters in CM. For instance, UE’s uplink SIR can be incorporated in CM2 in order to take into account the channel quality as a factor in the assigned rate for the UE. This leads to another CM that includes both UE buffer occupancy statuses as well as UE’s SIR which we refer to it as CM3. Details of CM3 calculation is provided in section A.2.

In addition, it is possible to execute VCUP algorithm without taking into account the assigned rate from NodeB. To distinguish, we define VCUP Case A and B as follow:

- Case A: VCUP scheduling based on NodeB assigned rate, UE available power, and buffer occupancy. Hence UE transmission rate is limited by NodeB assigned rate, UE available power, and buffer occupancy.
Chapter 2. Scheduling in Uplink

- Case B: VCUP scheduling based on UE available power, and buffer occupancy. In this case, UE transmission rate is only limited by UE available power and buffer occupancy, but not limited by NodeB assigned rate.

Extensive system level simulation results for VCUP algorithm using CM2 and CM3 with different VCUP cases and load conditions are presented and compared in Appendix A alongside benchmark scheduling algorithms.

Recalling from previous section, the main advantage of decentralized scheduling over centralized one in general is to reduce the level of overhead signalling required in centralized scheduling and increase spectral efficiency and reduce delay. This is also the motivation behind developing distributed scheduling such as VCUP to further exploit this advantage. However, VCUP is a semi-distributed scheduling and not fully distributed scheduling as UE is a decision maker but somehow limited to its serving NodeB.

2.5 Summary

In this chapter, first an overview on uplink packet scheduling in HSUPA system is presented. HSUPA system is used here as an example mainly because the importance of resource scheduling came to spot light and was more appreciated with the support of high data rate services in the evolution of UMTS standard to HSDPA (downlink) and HSUPA (uplink).

Uplink resource allocation methods in HSUPA can be categorized as centralized or decentralized in terms of the network location/node in which scheduling takes place. In centralized scheduling, resource allocation for all users resides in RNC. In contrast, NodeB controls scheduling and rate assignment for its own user terminals in decentralized scheduling. The main advantage of decentralized scheduling over centralized one is to reduce the level of overhead signalling required in centralized scheduling and hence increase spectral efficiency and reduce delay.

Two reference decentralized scheduling algorithms proposed by Nokia and Qualcomm are presented and will be used later in this thesis as benchmark for system performance comparison. In addition, a combined algorithm by Fujitsu called VCUP is presented. VCUP is a semi-distributed scheduling as UE is a decision maker but somehow limited to its serving NodeB.

In the next chapter, we present novel distributed scheduling algorithms in which UE solely decides on the uplink transmission rate instead of RNC or NodeB. Detailed simulation results are presented and compared with Nokia, Qualcomm and VCUP schedulers in chapter 4 and Appendix A.
Chapter 3

3 Developing Distributed Scheduling Algorithms

3.1 Introduction

The main motivation for developing fully distributed scheduling algorithms is to enhance uplink capacity and achieve efficient resource utilization. In chapter 2, we discussed the fundamentals of centralised/RNC scheduling, decentralised/NodeB scheduling and semi-distributed scheduling like VCUP and also presented some examples of referenced uplink scheduling algorithms.

We observe a trend in uplink scheduling from favouring centralised algorithms towards decentralized algorithms. In earlier 3GPP standards [4][5][6], the uplink scheduling and rate control resides in RNC. By providing the NodeB with appropriate measures, tighter control of the uplink interference is possible which in turn result in increased capacity and improved coverage.

Recent studies [3] show better performance for decentralized schedulers compared to centralized ones especially in terms of delay performance.

What we are investigating here is basically an extension to this trend leading to developing scheduling techniques we call distributed to distinguish from decentralized ones.

In simple words, one can say resource allocation decision has moved from RNC to NodeB if we compare centralized with decentralized scheduling. On this basis, semi-distributed scheduling techniques such as VCUP can be seen as (partially) moving resource allocation decision from NodeB towards UE aiming to further extend the advantage of fast scheduling and reducing signalling overhead and latency. However, VCUP is not fully distributed as Node-B remains an important element in the resource allocation by defining UE’s TFCS and CM. In this chapter we introduce novel and fully distributed scheduling algorithm based on rate back-off concept. Rate back-off is a similar concept to time back-off used for re-transmission in ALOHA access scheme [32]. The main idea is to grant UE complete control over its transmission rate based on instantaneous uplink interference at Node-B and hence being called distributed scheduling.
In the rest of this chapter, couple of such algorithms will be presented and discussed in details to show how we can execute this technique in scheduling. Extensive simulation results are presented in chapter 4.

3.2 Distributed Scheduling UniS Algo-1

3.2.1 Parameter definition and Initialization

Let us start with introducing two basic metrics used in our distributed scheduling algorithms i.e. Comparative Buffer (CB) and Capacity Indicator (CI).

CB is a comparative metric specific to each EU based on its buffer size and CI is an indicator for capacity/interference at Node-B which is defined as CI=Noise Rise Target (NRT) / RoT.

CB is a user-specific value but CI is a single cell-specific value, independent of UEs. Therefore, it can easily be signalled over broadcast channel across the cell coverage area with minimal signalling overhead. Let define $Q_{bR_i}$ as the Queue-based assigned Rate at UE_i, the lowest rate in the TFC subset available for UE_i which can empty its buffer $Q_i$ within one TTI.

$$Q_{bR_i} = f(Q_i / TTI)$$  \hspace{1cm} (3.1)

$f(Q_i / TTI)$ returns the closest higher rate in the TFC subset to the $Q_i / TTI$. Note that $Q_{bR_i} \in TFCS$.

We also define $C_{bR_i}$ as the Cell-based assigned Rate for UE_i. It will be used only if the cell is congested i.e. when CI is less than 1.

After receiving CI from serving Node-B, UE_i calculates its $C_{bR_i}$ using its current assigned rate $R_i$ as follows:

$$C_{bR_i} = Map(TFCS, CI, R_i)$$  \hspace{1cm} (3.2)

The $Map$ function returns a transmission rate from TFCS based on current rate $R_i$ and CI value.

Let assume TFC Subset has 7 different rates as $TFCS = \{8, 16, 32, 64, 128, 256, 384 \text{ kb/s}\}$ and $R_i = TFCS \lfloor k \rfloor$. Then:

$$Map(TFCS, CI, R_i) = TFCS \lfloor \max(k-(10-\text{floor}(CI*10)), 0) \rfloor$$  \hspace{1cm} (3.3)
Chapter 3. Developing Distributed Scheduling Algorithms

Note TFCS\(0\)=8kb/s is the minimum guaranteed rate in any circumstances if UE has something in its buffer to transmit. We also use two control parameters CB\(_{\text{Threshold}}\) and CI\(_{\text{Threshold}}\). CB\(_{\text{Threshold}}\) is the rate control parameter used to prevent UE’s buffer from over-flow.

CI\(_{\text{Threshold}}\) is the congestion control parameter which defines whether or not the serving cell’s load (or equally uplink interference) is in critical condition.

CB\(_{\text{Threshold}}\) and CI\(_{\text{Threshold}}\) should be initialised first. Just as an example here, we assume CB\(_{\text{Threshold}}=50\%\) and CI\(_{\text{Threshold}}=1\).

Also let initialize UE transmission rate to its \(QbR\) which is the lowest rate in TFCS which can empty the buffer next TTI.

3.2.2 Algo\(-1\) description

As mentioned before, distributed scheduling is completely taking place in UE, not in Node-B. The only contribution from Node-B is to send the updated CI and CB values to all UEs it serves at every scheduling instant. Each UE then decides on its transmission rate taking into account the cell congestion and its buffer status.

\[
R_{i(k+1)} = \min(QbR_i, CbR_i)
\]

Figure 3-1: Distributed scheduling Algo-1 flowchart
Chapter 3. Developing Distributed Scheduling Algorithms

Figure 3-1 shows the flowchart of this algorithm for UE, at scheduling instant \( k+1 \) assuming its previous rate as \( R(k) \). After initialization in the beginning, at scheduling instant \( k+1 \), for \( i \)th UE the following steps will construct the scheduling decision:

1. UE calculates its next transmission rate \( R_i(k+1) \) based on the cell status and its own buffer using the algorithm below:
   - If ( \( CI > CI_{Threshold} \) ), which means the cell has available capacity and it is not congested, then:
     - If ( \( QbR_i < R_i(k) \) ), which means the rate chosen by Queue is less than or equal to previous rate, then \( R_i(k+1) = QbR_i \).
     - If ( \( QbR_i \geq R_i(k) \) ), which means the UE is demanding for higher rate, then:
       - If ( \( CB > CB_{Threshold} \) ), which means the UE is probably in critical situation, then \( R_i(k+1) = QbR_i \).
       - If ( \( CB \leq CB_{Threshold} \) ), then \( R_i(k+1) = R_i(k) \).
   - If ( \( CI \leq CI_{Threshold} \) ), which means the cell is congested, then:
     - If ( \( QbR_i < R_i(k) \) ), which means the rate chosen by Queue is less than or equal to previous rate, then \( R_i(k+1) = \min(QbR_i, CB_R) \).
     - If ( \( QbR_i \geq R_i(k) \) ), which means the UE is demanding for higher rate, then:
       - If ( \( CB > CB_{Threshold} \) ), which means the UE is probably in critical situation, then \( R_i(k+1) = R_i(k) \).
       - If ( \( CB \leq CB_{Threshold} \) ), then \( R_i(k+1) = CB_R \).

2. UE then checks the \( R_i(k+1) \) in terms of its available transmit power. If it does not have enough power head room, it will choose the maximum available rate.

3. Under any circumstances, if UE has data and available power, it can autonomously transmit at minimum rate in TFCS, which is \( R_{min} = 8 \) kbps.

3.3 Distributed Scheduling UniS Algo-2

3.3.1 Parameter definition and Initialization

In the second distributed scheduling algorithm, CB signalling is cancelled. Instead, a normalized UE queue size is used, which is available at UE, eliminating the additional signalling overhead.
Chapter 3. Developing Distributed Scheduling Algorithms

Therefore in this algorithm, only common channel signalling for CI is required. Hence this algorithm only requires downlink common broadcast channel signalling.

Two metrics used in this scheduler are Normalized Queue (NQ) and CI as in Algo-1. NQ is the UE’s queue normalized to its maximum buffer size.

Similar to Algo-1, we use two control parameters namely NQ_Threshold and CI_Threshold. CI_Threshold remains as before, while NQ_Threshold is a control parameter which defines whether or not the UE is in critical situation in terms of its buffer occupancy. NQ_Threshold is a service-dependent parameter hence a single value for same group of users.

Same as in Algo-1, let initialize UE transmission rate to its $QbR$ which is the lowest rate in TFCS which can empty the buffer next TTI.

Also NQ_Threshold and CI_Threshold should be initialized first. Just as an example here, we assume NQ_Threshold= 50% and CI_Threshold=1.

3.3.2 Algo-2 description

In this algorithm, again, scheduling is completely taking place in UE, not in Node-B. However, in contrast to Algo-1, the only contribution from Node-B is to send one single value of updated CI to all UEs it serves at every scheduling instant. Instead of sending CB, normalized UE’s queue size (NQ) which is available at UE will be used. Therefore, each UE decides on its transmission rate taking into account the cell congestion and its buffer status. Figure 3-2 shows the flowchart of this algorithm for UE_i at scheduling instant $k+1$ assuming its previous rate as $R_i(k)$.
Chapter 3. Developing Distributed Scheduling Algorithms

Figure 3-2: Distributed scheduling Alg-2 flowchart

After initialization in the beginning, at scheduling instant $k+1$, for $i^{th}$ UE the following steps will construct the scheduling decision:

1. UE calculates its next transmission rate $R_i(k+1)$ based on the cell status and its own buffer using the algorithm below:

   - If ($CI > CI_{\text{Threshold}}$), which means the cell has available capacity and it is not congested, then:
     - If ($Q_bR_i < R_i(k)$), which means the rate chosen by Queue is less than or equal to previous rate, then $R_i(k+1) = Q_bR_i$.
     - If ($Q_bR_i \geq R_i(k)$), which means the UE is demanding for higher rate, then:
       - If ($NQ > NQ_{\text{Threshold}}$), which means the UE is probably in critical situation, then $R_i(k+1) = Q_bR_i$.
       - If ($NQ \leq NQ_{\text{Threshold}}$), then $R_i(k+1) = R_i(k)$.

   - If ($CI \leq CI_{\text{Threshold}}$), which means the cell is congested, then:
Chapter 3. Developing Distributed Scheduling Algorithms

- If \((QbR_i < R_i(k))\), which means the rate chosen by Queue is less than or equal to previous rate, then \(R_i(k+1) = \min(QbR_i, CbR_i)\).

- If \((QbR_i \geq R_i(k))\), which means the UE is demanding for higher rate, then:
  - If \((CB > CB_{\text{Threshold}})\), which means the UE is probably in critical situation, then \(R_i(k+1) = R_i(k)\).
  - If \((CB \leq CB_{\text{Threshold}})\), then \(R_i(k+1) = CbR_i\).

2. UE then checks the \(R_i(k+1)\) in terms of its available transmit power. If it does not have enough power head room, it will choose the maximum available rate.

3. Under any circumstances, if UE has data and available power, it can autonomously transmit at minimum rate in TFCS, which is \(R_{\text{min,TFCS}} = 8\) kbps.

3.4 Summary

Resource allocation decision making has moved from RNC to NodeB if we compare centralized with decentralized scheduling. In this regard, semi-distributed scheduling such as VCUP partially moves resource allocation decision making from NodeB towards UE aiming to further extend the advantage of fast scheduling and reducing signalling overhead and latency. However, VCUP is not fully distributed as Node-B remains an important element.

This chapter presents two novel scheduling algorithms developed based on rate back-off concept, namely Algo-1 and Algo-2, which are fully distributed in order to enhance uplink capacity and achieve efficient resource utilization. Algo-1 uses two basic metrics i.e. Comparative Buffer (CB) and Capacity Indicator (CI). CB is user-specific based on comparative UE’s buffer size, and CI is an indicator for uplink capacity at Node-B and hence is cell-specific. In Algo-2, CI metric is still used but CB is replaced with normalized UE queue size (NQ) metric which is locally available at UE and therefore eliminating the additional signalling overhead. As the result, overhead signalling in Algo-2 is minimized due to the fact that CI is a cell-specific value and can be sent over broadcast channel with minimal signalling overhead.

Extensive simulation results evaluating Algo-1 and Algo-2 are presented in chapter 4 alongside Nokia, Qualcomm and VCUP schedulers for comparison.
Chapter 4

4 System level simulations

4.1 Introduction

In chapter 2, main scheduling strategies as well as some referenced centralized and decentralized scheduling algorithms presented and discussed alongside recently developed semi-distributed scheduling VCUP. In chapter 3, we introduced novel and fully distributed scheduling algorithms using rate back-off concept. The main idea is for UE to gain complete control over its transmission rate based on instantaneous uplink interference at Node-B and hence we called it distributed scheduling.

In this chapter, we evaluate the performance of distributed scheduling algorithms and compare them with standard schedulers presented in chapter 2 by means of system level simulations.

To evaluate performance and quantify the gain, HSUPA system level simulator has been developed, verified against standard and used as the platform. In this chapter, firstly, an overview on the system level simulation platform is given in section 4.2, including the simulation methodology and main features, followed by validation results in section 4.3. Comprehensive system level simulation results for benchmark Qualcomm's decentralised scheduling algorithm, Nokia's RUF-based algorithm, as well as results from VCUP and distributed scheduling algorithms are presented and compared in section 4.4.

4.2 Overview of the Simulation Platform

4.2.1 Simulation Topology

In the system level simulation platform used for uplink UMTS, fully dynamic simulation approach is employed as the simulation methodology. At the beginning of each simulation, UEs are created and remain active for the entire simulation episode. These users are randomly and
uniformly distributed over a network of tri-sectored cells. The simulated network is constituted of 57 sectors (19 Base Stations with 3 sectors) as illustrated in Figure 4-1.

Figure 4-1: Network deployment

4.2.2 Main Features

The simulation platform is based on an object-oriented design and implemented using C++. The main features of the simulation platform are as follows:

- Grid of 3-sector sites
- 3GPP standard antenna pattern
- 3GPP standard propagation models
- 3GPP standard traffic models
- Full buffer option support
- Mixed traffic support
- Mixed propagation conditions support
- Soft handover
- Uplink inner and outer-loop power control

For each time slot, channels between Users and BSs are updated and fast power control is performed. Then at each TTI, data packets are generated according to traffic models. Also packet transmission and reception is performed at TTI time scale. On a longer time scale, scheduling is performed at each scheduling period.

4.3 Simulation Validation

4.3.1 Simulation Methodology

To validate the simulation platform, baseline simulations have been carried out to reproduce reference simulation results of 3GPP Release-99 system as presented in [3]. The simulation
results and those presented in [3] are both included in this section. Same simulation parameters as in [3] have been used to ensure valid comparison. Baseline simulation assumptions are:

- 19 3-sectored cells with wrap-around
- Full buffer for users
- No TFC control
- All UEs transmit with 384 kbps
- AWGN channel
- Pedestrian mobility with speed 3km/h
- Close loop inner and outer loop power control

Figure 4-2 shows a snap shot example of UE distribution in the simulation. As mentioned, users are randomly distributed in the simulated network area and each cell has the same number of UEs.

### 4.3.2 RoT Performance

Figure 4-3 illustrates the average RoT as a function of the number of UEs per cell. It can be seen that as the number of UEs increases, the RoT increases. Clearly, simulation results of the simulation platform used fit very well with those of 3GPP system presented in [3].
Chapter 4. System level simulations

4.3.3 UE Throughput Performance

Figure 4-4 shows the scatter plot of the user throughputs for 5, 10 and 15 users per cell as a function of the best downlink path loss in 3GPP system. It can be seen that as the number of UEs increases, the cell coverage decreases. This phenomenon is often known as cell breathing.

Figure 4-5 shows the scatter plot of the user throughputs for 5, 10 and 15 users per cell as a function of the best downlink path loss for our simulation platform. Comparing Figure 4-4 and Figure 4-5, there is slight difference in the scatterplot. In case of 15 UE per cell, users with minimal transmission rate are allowed in our simulator, which increases the possibility of higher
path loss. Similarly, in case of 5 UE per cell, users with highest transmission rate may occur in locations with lower path loss.

![User throughput vs. downlink path loss](image)

**Figure 4-5: User throughput vs. downlink path loss from our simulation platform**

### 4.4 Scheduling Simulation Results

#### 4.4.1 Objectives

Numerous simulation runs have been carried out to investigate distributed scheduling algorithms presented in chapter 3 and to compare its performance under different load conditions including 10 UEs per cell, 15 UEs per cell and hotspot scenarios with the results from conventional schedulers.

Since the trend of results observed under different load conditions remain the same, here only detailed simulation results for 10 UEs per cell simulations are presented. Complete set of simulation results including 15 UEs per cell and hotspot scenarios are presented in Appendix A.

The simulation cases are built upon different scheduling algorithms namely Nokia RUF-based scheduler (see section 2.2), Qualcomm decentralised scheduler (see section 2.3), VCUP (see section 2.4), and UniS algo-1 and UniS algo-2 presented in chapter 3.

#### 4.4.2 Simulation Cases

Six simulation cases have been considered as basis for performance comparison:
Chapter 4. System level simulations

1. Nokia RUF-based scheduler
2. Qualcomm NodeB scheduler
3. VCUP scheduler (CM2 Case B with SHO combining)
4. VCUP scheduler (CM2 Case B without SHO combining)
5. UniS algo-1 scheduler
6. UniS algo-2 scheduler

In all these cases, the system is HSUPA and simulation parameters are kept exactly same to allow fair comparison. Maximum cell load (RoT threshold) for all the simulation runs set to 5.3 dB. The only difference between these cases is the scheduling algorithm used. Complete set of simulation parameters is presented in Appendix A, Table A.1.

For VCUP scheduler we present two cases with and without SHO combining. In case of SHO combining, UE uses the CM sent by all three NodeB’s in which UE is in SHO with. UE then uses the averaged CM to find its transmission rate. On the contrary, in case of without SHO combining, UE only uses CM sent by the serving NodeB and ignores CMs from other NodeB’s.

Also here, The CM is based on UE buffer occupancy (i.e. CM2) and VCUP case B as explained in section 2.4.2. This is due to the fact that CM2 has shown better performance compared with CM3. Complete range of simulation cases and results are provided and analysed in Appendix A.

4.4.3 Analysis

Figure 4-6 shows the Probability Distribution Function (PDF) of cell RoT and similarly Figure 4-7 shows the Cumulative Distribution Function (CDF) of cell RoT. It can be seen that in all simulation cases, the average RoT is maintained below the target 5.23 dB. Defining RoT outage probability as the percentage of RoT occurrence above target, it can be observed from Figure 4-7 that RoT outage in all cases is less than 5%. It also shows that RoT outage is slightly more in case of VCUP and distributed algorithms Algo-1 & Algo-2. Nevertheless, they manage to maintain RoT well below the target 5.23 dB.

Figure 4-8 and Figure 4-9 show the PDF and CDF of system total queue size respectively, at the end of each TTI. Total queue size is the summation of remaining data (bytes) in the buffer of all active UEs at the end of each transmission interval. Comparing the performance curves of schedulers in Figure 4-9, it appears that UniS algorithms outperform others by 30% to 50% and manage to empty UE buffers far quicker than other scheduling algorithms including benchmark Qualcomm and Nokia ones. Again, both benchmark algorithms seem to perform narrowly close.
Chapter 4. System level simulations

Figure 4-6: PDF of average Cell RoT

Figure 4-7: CDF of average Cell RoT
Figure 4-8: PDF of system total queue size

Figure 4-9: CDF of system total queue size
Figure 4-10: PDF of packet delay in number of TTIs

Figure 4-11: CDF of packet delay in number of TTIs

Figure 4-10 and Figure 4-11 show the PDF and CDF of mean packet delay in number of TTIs respectively. Comparing the delay performance for different schedulers in Figure 4-11, VCUP with SHO-combining seems to be worse than others. All other schedulers manage to keep user’s packet delay below 50 TTIs at more than 95% of times.
Chapter 4. System level simulations

Figure 4-12: OTA throughput vs. distance from cell site

Figure 4-12 and Figure 4-13 illustrate throughput performance versus distance from cell site. Throughput is presented in two forms; Over The Air (OTA) and Service throughput. OTA throughput is the summation of transmission rates (kbps) from all active users while Service throughput is the summation of successfully received (error-free) data.

It clearly shows that UniS scheduler **Algo-1** outperforms other schedulers significantly. It provides almost double OTA throughput in the short range (up to 1/3 of cell range) and similar throughput in the long range (cell border).

Same behaviour can be observed in Figure 4-14 and Figure 4-15 illustrating the buffer occupancy and mean packet delay, respectively, versus distance from cell site. Again, UniS scheduler **Algo-1** outperforms other schedulers, followed by UniS scheduler **Algo-2** in second place.

UniS scheduler **Algo-1** provides far less buffer occupancy (Figure 4-14) compared to benchmark Qualcomm and Nokia schedulers throughout the cell and far less packet delay (Figure 4-15) in the short-medium range.
**Chapter 4. System level simulations**

**Figure 4-13: Service throughput vs. distance from cell site**

**Figure 4-14: Buffer occupancy vs. distance from cell site**

41
4.5 Summary

After numerous rounds of system simulation runs to investigate performance of uplink scheduling algorithms presented in chapter 3 and to compare under different load conditions including 10 UEs per cell, 15 UEs per cell and hotspot scenarios, following conclusions can be drawn:

- In all simulation cases, the average RoT is maintained below the target 5.23 dB. Defining RoT outage probability as the percentage of RoT occurrence above target, it is observed that RoT outage in all cases is less than 5%.

- All schedulers manage to have less than 50 TTI packet delay at more than 95% of times, except for VCUP with SHO-combining.

- It is observed that UniS Algo-1, the novel distributed scheduling technique introduced in this thesis, can achieve superior performance compared against all other types of algorithms presented in chapter 2 as benchmark, in terms of throughput, packet delay, buffer occupancy etc. UniS Algo-2 comes as second best.
Chapter 4. System level simulations

- When the system loading increases (e.g. 15 UEs per cell), it is observed that UniS Algo-1 and Algo-2 still achieve superior performance in general, but with less gain margin compared with benchmark algorithms, such as Qualcomm or Nokia scheduling algorithm.

- In the simulation with hot spots, similarly, UniS Algo-1 outperforms all other algorithms in terms of throughput, buffer occupancy etc., and again when the system loading increases, the gain margin decreases. Hot spot results are presented in Appendix A.

- It is observed that for those users close to the cell site, UniS Algo-1 could achieve very good performance in terms of throughput, packet delay, buffer occupancy etc, but not so good for those users who are far away.

- The advantage of Algo-2 over Algo-1 is in the signalling overhead. Algo-2 does not require any user-specific signalling as opposed to Algo-1. The signalling overhead in Algo-2 is minimal, a cell-specific value sent over broadcast channel prior to each scheduling instant.

- For semi-distributed VCUP scheduler, CM combination for SHO UEs does not show any gain, in comparison with CM without SHO-combining.

In summary, the results show significant improvement in uplink scheduling by deploying distributed algorithms such as Algo-1 and Algo-2. It extends the advantage of decentralised scheduling further by transferring more control to UEs to decide on their uplink transmission rate instead of RNC or NodeB. We observed that distributed schedulers like Algo-1 outperform conventional uplink schedulers significantly.

Now, there are some important questions to answer. Is distributed scheduling the answer to optimum uplink scheduling? Is there possibly an inherent conceptual shortcoming from conventional schedulers which also affects distributed scheduling and prevents reaching maximum capacity? What is the upper-bound scheduling performance anyway?

In chapter 5, we present an important fact in any scheduling technique to date: A considerable proportion of RoT comes from inter-cell interference which NodeB has little knowledge about and control upon. We provide a detailed look into this problem and highlight the importance of Intercell interference control as the key factor in future radio resource management.

The summit of this thesis comes in chapter 6 which addresses this problem by introducing a novel scheduling technique called Load Matrix.
Chapter 5

5 Importance of Intercell Interference Control

5.1 RoT as resource utilization indicator

Uplink cell capacity in interference-limited systems such as UMTS is basically limited by the total received power at the base station. As the uplink load increases, user terminals have to increase their transmit power substantially to overcome the increased interference level at the base station [18]. Due to the fact that the transmit power of user terminals is limited, total received power at base station actually limits the uplink capacity.

In interference-limited systems, RoT of a cell is a good indicator of cell load. Figure 5-1 shows typical cumulative distribution function of RoT at base station. The ideal performance in terms of interference management would be to keep actual RoT as close to the threshold line as possible resulting in a step-function RoT shape. In Figure 5-1, the area marked by “A” captures instances when resources have been allocated more than it should, and “B” marks opposite instances when there are unused resources that could be allocated to users.

Recall simulation results in chapter 4, looking at Figure 4-7 in particular, one can see that all scheduling algorithms come short to utilize cell RoT properly. Why this RoT behaviour occurs in all cases? Does it indicate a generic conceptual problem?

In decentralized scheduling, each base station assigns radio resources (i.e. transmission rate and time) to its users until the estimated RoT reaches a predefined target value, $RoT_{target}$. $RoT_{target}$ is usually a fixed target value set by the network controller to maintain uplink interference level [3].

It is true that Intracell interference is generated by own cell users and it is manageable by serving base station. However the main shortcoming for conventional decentralized scheduling in general becomes more visible in a multi-cell scenario where a considerable proportion of RoT is intercell interference and base station has little knowledge about and control upon. Similarly, distributed
Chapter 5: Importance of Intercell Interference Control

Scheduling methods including UniS algorithms, in which mobile users take control of the scheduling from serving base station, also suffers from this shortcoming.

By intercell interference here, we mean intercell multiple-access interference in general i.e. any signal received by a base station (regardless of its orthogonality properties) coming from those users which belong to other cells.

![CDF graph](image)

**Figure 5-1: Resource Utilization interpretation of RoT**

The relationship between cell RoT and intercell interference is given in (5.1). The RoT of cell j ($RoT_j$) is defined as the total in-band received power at base station j ($BS_j$) over thermal noise.

Let $N'$ be the received noise power in $BS_j$, $P_i$ be the transmission power of user $i$ and $G_{ij}$ be the channel gain from user $i$ to $BS_j$. For M active users in the network, $RoT_j$ can then be written as:

$$RoT_j = \left( \sum_{i=1, i \text{ belongs to } BS_j} P_i G_{ij} + \sum_{i=1, i \text{ doesn't belong to } BS_j} P_i G_{ij} + N' \right) / N'$$  \hspace{1cm} (5.1)

For simplicity, we have not considered soft hand over here meaning the user is assumed connected only to one base station at a time.
Lack of proper RoT utilization is not the root cause of the entire problem. The scope of intercell interference problem in uplink scheduling becomes more evident when we look at RoT fluctuation in time in the following section.

5.2 RoT fluctuation in time

So far, we identified that the main challenge in achieving efficient uplink scheduling comes from intercell interference and how to deal with it. Figure 5-2 reveals another phenomenon in RoT behaviour with sever degrading impact on system performance. It shows typical RoT fluctuation in a cell due to intercell interference using a conventional scheduler. In one scheduling instant (scheduling is assumed to take place every 10 TTI here), RoT of a cell dramatically increases to well above the threshold and in the following scheduling instant drops to well below the threshold. This again highlights the fact that lack of information about neighbouring cells poses negative impact in terms of interference outage which in turn increases the probability of packet errors.

![RoT fluctuation in time for a typical cell](image)

Figure 5-2: Typical RoT fluctuation (in time) in conventional scheduling due to intercell interference

Interestingly enough, Figure 5-2 shows RoT fluctuation under normal condition in a typical multi-cell scenario. To highlight the importance of intercell interference problem and to give an idea
about severity of its impact in the overall interference outage performance and resource utilization, we set up an extreme simulation scenario.

In this scenario, all the cells have same traffic distribution and interference condition (i.e. identical user distribution per cell). In each cell 10 users with full buffer are waiting for transmission. Users are randomly and uniformly distributed over one cell and then replicated with same pattern over other cells. The simulated network is constituted of 19 Omni-directional cells with wrap-around as shown in Figure 5-3. The RoT target is set to 5.2 dB equal to 70% load factor. Scheduling is decentralized, taking place individually in each base station every 10 TTI that is every 100 ms. Assumption of full buffer occupancy for users is taken to model the network behaviour under heavy traffic load. We have chosen this extreme scenario to see how far the problem of intercell interference can go in its extreme situation as the worst case scenario.

![Cell layout and UE distribution](image)

Figure 5-3: Extreme scenario with identical cells

The decentralized scheduling algorithm in [8] is used which allocates resources individually in each cell.

The RoT result is overwhelmingly unstable. Figure 5-4 shows the RoT fluctuation of a cell in this scenario for a period of 20 seconds. A closer look is provided in Figure 5-5 for 1 second period.
Looking at Figure 5-5, at TTI=1140, RoT level is way below the RoT target, so scheduler decides to allocate resource to more users unaware that all other cells will do the same thing. The consequence of this decision is much higher RoT than expected in the next scheduling interval. At TTI=1150, the opposite happens; RoT level prior to scheduling instant is way above the target RoT so scheduler severely decreases the amount of allocated resources unaware that all other cells will do the same thing. This phenomenon continues and RoT takes a pulsy shape far from RoT target, as shown in Figure 5-5.

As mentioned earlier, this is the worst case scenario in respect to the intercell interference problem. Nevertheless, some degree of fluctuation such as in Figure 5-2 has been always observed in RoT performances regardless of the scheduling type or algorithm being used. RoT fluctuation obviously gets worse when the traffic load increases. We have assumed an extreme scenario where the load in neighbouring cells varies in a synchronised way which is not realistic but it is a good way to model heavy load without increasing number of users.

In reality however, it is expected that neighbouring cells are all facing very high traffic in peak hour (i.e. users with full buffers waiting for transmission) at the same time. Also it is likely to have similar traffic behaviour in neighbouring cells since they are geographically close and therefore may have users with similar social environments and activities.
This case study clearly shows that intercell interference is a crucial factor which cannot be ignored in the scheduling process for future wireless cellular systems.

![RoT fluctuation in time for a typical cell (1 second)](image)

**Figure 5-5: RoT fluctuation in extreme scenario (1 second observation)**

### 5.3 Summary

In this chapter, we present main shortcomings in conventional uplink scheduling which is also inherited by distributed scheduling. First, the ideal performance in terms of interference control is shown to be step-function RoT at RoT target level. Areas where resources are not fully utilized or over used (resulting in higher error rate and less throughput) are explained. Second, the importance of inter-cell interference in overall cell capacity is shown. A considerable proportion of RoT comes from inter-cell interference which NodeB has little knowledge about and control upon. Last but not least, RoT fluctuation in a cell due to intercell interference using conventional scheduling is explained. In one scheduling instant, RoT of a cell dramatically increases to well above the threshold and in the following scheduling instant drops to well below the threshold. Detailed look into cell RoT fluctuation in time is presented in couple of simulation cases highlighting the importance of Inter-cell interference control as the key factor in future radio resource management.

In the rest of this thesis, we address the influence of multi-cell interference on overall radio resource utilisation and propose an innovative and novel technique to uplink scheduling, setting a
new direction for future research on resource scheduling strategies in a multi-cell system. We propose a technique called Load Matrix (LM) which facilitates joint management of interference within and between cells for allocation of radio resources. The idea of Load Matrix is similar to interference matrix used for frequency planning in TDMA systems [33]. It consists of corresponding load factors of each user in every cell throughout the network. Load Matrix technique towards uplink scheduling in thoroughly presented and discussed in the following chapter.
Chapter 6

6 Load Matrix Scheduling Technique

6.1 The concept and the constraints

In any challenge, defining and understanding the problem is always a vital step towards finding the best solution and radio resource allocation is no exception. The aim of resource allocation in wireless cellular systems is to assign radio resources to individual users in a way to achieve maximum system capacity whilst meeting the required quality of service. Before introducing Load Matrix, first we formulate the resource allocation problem in wireless cellular system and show that it is in fact a Non deterministic Polynomial time (NP)-hard problem [34].

Let us consider a basic uplink scenario where resource allocation is down to assigning transmission rate and time to individual users with the objective of throughput maximization. To analyze the problem, we begin with the single cell scenario and then extend the conclusion to the multi-cell case. Without loss of generality, we assume that transmission rates are chosen from a limited set of rates.

Let $S_i$ denote Candidate Rate Set (CRS) of user $i$, which includes all the allowed transmission rates for the user to choose from. Rate "0" is always included in $S_i$, and will be chosen if the user is not scheduled to transmit in the current scheduling instant.

We treat transmission rates in different CRSs as different items even if they have the same rate value:

$$S_i \cap S_j = \emptyset \quad \forall i \neq j$$  \hspace{1cm} (6.1)

Let $S$ denote the union of all the CRSs from $S_1$ to $S_n$, and $M$ is the total number of users in the cell sharing the radio resource pool. Choosing an element $e$ from set $S$ is an assignment action, which means allocating a specific transmission rate to a particular user. Apparently, each assignment action generates a certain amount of throughput while consumes some amount of the cell capacity. We use binary variable $x_e$ to indicate whether element $e$ is chosen or not (1 for ‘Yes’ and
Chapter 6. Load Matrix Scheduling

0 for ‘No’). \( r_e \) and \( c_e \) denote the generated throughput and consumed cell capacity respectively if element \( e \) is chosen. \( r_e \) is equal to the transmission rate itself, whereas \( c_e \) can be interpreted differently, e.g. as consumed BS transmit power or generated load factor [35], depending on the system type.

Using above terms and definitions, the Single Cell Radio Allocation Problem (SCRAP) can be described as follows: given a particular system snapshot (cell capacity, user location, propagation and traffic status etc.), how to choose elements from set \( S \) in each scheduling instant so as to achieve maximum system throughput, subject to the following two constraints:

\( C1 \): The aggregated cell capacity consumption of all the chosen elements from \( S \) should be less than the total available cell capacity \( C \).

\( C2 \): for each CRS \((S_1, \ldots, S_M)\), only one element is chosen.

Mathematically, SCRAP can be formulated as follows:

\[
\text{maximize:} \quad r = \sum_{e \in S} r_e x_e \\
\text{subject to:} \quad \sum_{e \in S} c_e x_e \leq C \quad (6.2)
\]

\[
\sum_{e \in S} x_e = 1, \quad i = 1, \ldots, M \quad (6.3)
\]

\[
x_e = \begin{cases} 
1 & \text{if 'e' is chosen} \\
0 & \text{if 'e' is not chosen} 
\end{cases}, \quad e \in S 
\quad (6.5)
\]

Theorem: The SCRAP is NP-hard.

Proof: We show SCRAP is in fact the Multi-Choice Knapsack Problem (MCKP) which has been proven to be NP-hard [36]. MCKP can be expressed as follows:

Given a knapsack, an item set, and a partition of the item set into a number of subsets, how we choose items so as to maximize the total profit from all the chosen items, while the aggregate weight of all the chosen items is less than the allowed weight bearing of the knapsack. There is also a condition that only one item is chosen from each item subset.
Chapter 6. Load Matrix Scheduling

SCRAP can be mapped to MCKP by regarding the knapsack capacity as the available cell capacity, the item set as the set \( S \), and the item subsets as CRSs \( (S_1, \ldots, S_M) \). Also \( r_e \) and \( c_e \) can be regarded as the profit and weight of element \( e \) respectively:

<table>
<thead>
<tr>
<th>SCRAP</th>
<th>MCKP</th>
</tr>
</thead>
<tbody>
<tr>
<td>available cell capacity &quot;C&quot;</td>
<td>Knapsack capacity</td>
</tr>
<tr>
<td>union of all CRSs &quot;S&quot;</td>
<td>the item set</td>
</tr>
<tr>
<td>Candidate Rate Sets &quot;( S_1, \ldots, S_M )&quot;</td>
<td>the item subsets</td>
</tr>
<tr>
<td>transmission rate &quot;( r_e )&quot;</td>
<td>profit of item</td>
</tr>
<tr>
<td>load factor &quot;( c_e )&quot;</td>
<td>weight of item</td>
</tr>
</tbody>
</table>

In MCKP all the components of the problem including knapsack capacity, items, profits, etc. are known and the question is which items to put in the knapsack. Similarly in the SCRAP, the only unknown parameter in equations \( (6.2) \) ~ \( (6.5) \) is \( x_e \). \( r_e \) is known because it is equal to the transmission rate itself. In CDMA uplink systems, \( c_e \) and \( C \) can be interpreted as the load factor of the given user (e.g. 10%) and the maximum load threshold of the cell (e.g. 70%) respectively. Then the following method can be used to calculate \( c_e \) for each element \( e \) before the rate allocation actually takes place:

1. Find out the target SINR at the receiver based on the given data rate and Block Error Rate (BLER). These target SINRs can be obtained from the BLER versus SINR performance curves in the physical layer (also known as link to system interface curves) [3].

2. This target SINR can then be transformed into the load factor: \( c_e = \frac{\text{SINR}}{1 + \text{SINR}} \) [34].

In single-cell scenario considered, no intercell interference exists. Therefore the available cell capacity \( C \) can be regarded exactly the same as the given cell capacity, which is assumed to be a fixed and pre-known value in our case. Furthermore, CRSs \( (S_1 \text{ to } S_M) \) are pre-known sets based on the system restrictions, power headroom and queue status of individual users.

Therefore, SCRAP is a type of Multi-Choice Knapsack Problem and NP-hardness of it follows by a trivial transformation from the MCKP.

We now consider the multi-cell case of the problem (MCRAP), where the interested network area is covered by \( N \) cells, each of which includes \( M_j \) users and \( \sum_{j=1}^{N} M_j = M \). In this case, \( c_e \) and \( C \) are upgraded to \( c_{e,j} \) and \( C_j \) where the subscript \( j \) represents the cell index:

\[
\sum_{e \in S} c_{e,j} x_e \leq C_j \quad \forall j = 1, \ldots, N
\]
In which \( c_{ij} \) is the consumed capacity from cell \( j \) if element \( e \) is chosen.

The expression of MCRAP is very similar to SCRAP except that constraint \( C_i \) should be replaced by the following \( C_i' \) in order to take all the cells into account:

\[ C_i' : \text{For any cell } j \ (j=1, \ldots, N), \text{ the total cell capacity consumption by all the chosen elements from } S \text{ should be less than the total available cell capacity } C_j. \]

In fact, SCRAP was the simplest case of MCRAP where the number of cells is 1. Consequently, if SCRAP is not solvable by a polynomial time algorithm, neither is MCRAP and therefore MCRAP is also NP-hard.

As mentioned earlier, the main challenge in resource allocation in a multi-cell system comes from intercell interference and how to control it. In uplink scheduling, the basic problem is to assign appropriate transmission rate and time to all active users in such a way that results in maximum radio resource utilization across the network whilst satisfying the QoS requirements of all the users. Amongst other constraints, another important factor in the resource allocation is the user's transmit power. For a network of \( M \) users and \( N \) cells the constraints to be satisfied are:

\[ C_1 : \text{For active user } i \text{ in the network, its transmit power } P_i \text{ must be maintained in an acceptable region defined as } 0 \leq P_i \leq P_{i,\text{max}} \quad i \in \{1, \ldots, M\} \quad (6.7) \]

\[ C_2 : \text{The total received power at base station should be kept below a certain threshold for all } N \text{ base stations in the network.} \]

Without loss of generality, we use RoT as defined in HSUPA [3] to represent the interference constraints. \( \text{RoT}_j \) is the total in-band received power at the base station \( j \) (BS, j) over thermal noise. Let \( N' \) be the receiver noise power at base station, \( P_i \) represent the transmission power of user \( i \) and \( G_{ij} \) be the channel gain from user \( i \) to BS \( j \). For \( M \) active users in the network, \( \text{RoT}_j \) can be written as

\[ \text{RoT}_j = (N' + \sum_{i=1}^{M} P_i G_{ij}) / N' \quad (6.8) \]

In this case \( C_2 \) can be formulated as

\[ \text{RoT}_j \leq \text{RoT}_{\text{target}} \quad \forall j \in \{1, \ldots, N\} \quad (6.9) \]

where \( \text{RoT}_{\text{target}} \) is assumed to be a fixed target value set by the network controller to maintain the uplink interference level.
Chapter 6. Load Matrix Scheduling

$C_j$: For each user, depending on the transmission channel and speed, each rate $k$ has a specific $\text{SINR}_{\text{target},k}$. $\text{SINR}_{\text{target},k}$ is the signal to noise plus interference ratio required at the serving base station $j$ if rate $k$ is to be assigned to the user in order to achieve a given Block Error Rate. Rate $k \in \{1, \ldots, K\}$ is considered the highest rate acceptable (and therefore is the preferred rate) for user $i$ with serving base station $j$ if $\text{SINR}_{\text{target},k}$ is the highest SINR that can be achieved considering both $C_j$ and $C_2$

$$\text{SINR}_{i,j} \geq \text{SINR}_{\text{target},k} \quad i \in \{1, \ldots, M\}, \quad k \in \{1, \ldots, K\} \quad (6.10)$$

6.2 Load Matrix definition

Load Matrix (LM) database can be regarded as a storage matrix containing the load factors of all active users in the network. LM scheduling can be implemented in both centralized and decentralized manner. In a decentralized LM scheduling, base stations should implement identical LM database.

Here for simplicity, we only present the centralized LM scheduling where a centralized scheduler assigns radio resources to all the users in the network. Figure 6-1 illustrates an example of LM scheduling implementation based on the proposed system architecture for the 3rd Generation Long-Term Evolution (3G LTE) [37]. A central node called Access Core Gateway (ACGW) acts as network controller, connecting NodeBs to external network. In this example, LM scheduler is located in ACGW and scheduling is centralized.

![Figure 6-1: Centralized LM scheduling in a 3G LTE system](image)
Chapter 6. Load Matrix Scheduling

We assume the averaged channel gain (averaged over the scheduling period) from users to base stations is known to scheduler prior to rate assignment. In a network of $M$ users and $N$ cells, $LM_{ij}$ is the load factor [35] contributed by user $i$ in cell $j$

$$LM_{i,j} = \frac{P_i G_{i,j}}{N'} + \sum_{a=1}^{M} P_{a} G_{a,j}$$  \hspace{1cm} (6.11)$$

From the $LM_{ij}$ values stored in column $j$ of LM database, RoT of cell $j$ can be written as

$$RoT_j = \frac{1}{1 - \sum_{i=1}^{M} LM_{i,j}}$$  \hspace{1cm} (6.12)$$

Note that $RoT_j$ obtained from (6.12) is identical to $RoT_j$ definition given in (6.8). Let BS $j$ be the serving base station for user $i$ which controls its power and $G_{ij}$ be the overall channel gain from user $i$ to BS $j$ averaged over scheduling period. $SINR_{ij}$ can be written as

$$SINR_{i,j} = \frac{P_i G_{i,j}}{N' RoT_j - P_i G_{i,j}}$$  \hspace{1cm} (6.13)$$

By rearrangement of (6.13), $P_{i,k}$ (minimum required transmit power for user $i$ for satisfying $k^{th}$ rate) can be found from the following equation provided $RoT_j$ at the end of this rate assignment will not exceed $RoT_{target}$

$$P_{i,k} = \frac{N' RoT_{target} \cdot SINR_{target,k}}{G_{i,j} (1 + SINR_{target,k})}$$  \hspace{1cm} (6.14)$$

Note that in (6.14), the constraint $C_2$ is already satisfied by considering $SINR_{target,k}$. The $P_{i,k}$ is valid only if it satisfies constraint $C_1$ which states the maximum transmit power.

Additionally, $P_{i,k}$ value must satisfy the constraint $C_2$ which takes into account the impact of assignment of the highest applicable $k^{th}$ rate to user $i$ on intercell interference. This ensures the intercell interference caused by user $i$ in other cell $n \in \{1, ..., N \mid n \neq j \}$ using $k^{th}$ rate does not increase other cells’ RoT values beyond their $RoT_{target}$.

If all the above constraints are satisfied for $k^{th}$ rate, next step is to update $LM_{in}$ for all the elements in row $i$ of Load Matrix. $LM_{in}$ is the load factor imposed by user $i$ in cell $n$ using $k^{th}$ rate defined:

$$LM_{i,n} = \frac{P_{i,k} G_{i,n}}{N' + \sum_{a=1}^{M} P_{a} G_{a,n}}$$  \hspace{1cm} (6.15)$$
From (6.12), one can estimate the new RoT for all other cells and check if the constraint $C_2$ has been satisfied. If so, the $k^{th}$ rate (the highest acceptable rate) will be assigned to the user $i$, otherwise the same process is repeated for lower rates starting with $(k-1)^{th}$ rate.

After the first round of rate assignment to all the users, LM elements are updated and new RoT is calculated for each cell using (6.12). This is necessary because (6.14) and (6.15) are valid only if RoT is close to the $RoT_{target}$. As the rates assignment is an NP-hard problem (see section 6.1), it is not possible to exactly achieve $RoT_{target}$ for all the cells in the first round of rates assignment. This requires additional rounds (which we refer to it as iterations) of rate/power adjustments in order to minimize the difference between a cell RoT and its $RoT_{target}$. In other words, $P_{lk}$ is iteratively adjusted in (6.14) and then (6.15) by replacing $RoT_{target}$ with updated RoT from LM after each round of rates assignment. This check is an important step ensuring low probability of interference outage by keeping RoT just below $RoT_{target}$ and at the same time increasing resource utilization with high cell RoT values.

Obviously, the number of iterations depends on the difference between RoT and $RoT_{target}$ at the end of each scheduling process. Nevertheless, the simulation results presented in the following section are with no iteration and yet the difference between RoT and $RoT_{target}$ was found to be negligible. It should be noted that if a user is not in the full buffer status, the maximum rate index $K$ in (6.10) is limited to a rate that would result in emptying the buffer at the next scheduling instant.

Cell throughput per bandwidth (bps/Hz/cell) is often taken as resource utilization measure. However, there is a trade off between maximum cell throughput and fairness amongst users [38]. Priority functions are used to rank users in the scheduling process and make a balance between cell throughput and fairness. Commonly used priority functions are Round Robin, SINR (also called Max C/I), Proportional Fair (see e.g. [3][7][9]) and also most recently introduced Score-Based [39]. The Round Robin tries to maximize fairness amongst users regardless of a user channel condition and therefore results in poor throughput performance. Max C/I, on the other hand, ranks users in terms of their channel quality aiming for maximizing cell throughput at the expense of fairness. Both Proportional Fair and Score-based priority functions perform better than Round Robin in terms of throughput and better than Max C/I in terms of fairness.

Load Matrix technique, provides a generic solution for resource scheduling that does not preclude any priority function and can be combined with any of them. However, priority function has major impact on overall system performance for any scheduling technique including the LM. Here we introduce a priority function based on a user’s load vector that takes into account both its intra and intercell impact on the network. It is evident that giving priority to a user with better channel condition increases the cell throughput but in a multi-cell network it could have severe
impact on the throughput of other cells. This fact is considered here by defining Global Proportional Priority function as

\[
\text{priority}_i = \frac{G_{ij}}{\sum_{n=1, n \neq j} G_{in}} \quad \forall i \in \{1, ..., M\}
\]

where \(G_{ij}\) is the total channel gain from user \(i\) to BS \(j\) averaged over the scheduling period. The LM technique tries to maximize network capacity through inter and intracell interference management. Table 6.2 summarises LM technique used for rate assignment in a multi-cell wireless system.

**Table 6.2: LM Technique; scheduling procedure summary**

<table>
<thead>
<tr>
<th>Initialize LM([i, j]) to zero;</th>
</tr>
</thead>
<tbody>
<tr>
<td>for cell (j = 1: N)</td>
</tr>
<tr>
<td>for user ([i]) in cell ([j])</td>
</tr>
<tr>
<td>set priority ([i, j])</td>
</tr>
<tr>
<td>end</td>
</tr>
<tr>
<td>sort all users in cell ([j]) according to priority ([i, j])</td>
</tr>
<tr>
<td>end</td>
</tr>
<tr>
<td>for assignment round = 1: Number of users per cell</td>
</tr>
<tr>
<td>for cell (j = 1: N)</td>
</tr>
<tr>
<td>for the highest priority user ([i]) in cell ([j]) if LM([i, j]) = 0</td>
</tr>
<tr>
<td>for Rate(_{\text{index}}) = (K: -1: 1)</td>
</tr>
<tr>
<td>get P(<em>{ij}) for Rate(</em>{\text{index}})</td>
</tr>
<tr>
<td>(condition 3 is already satisfied)</td>
</tr>
<tr>
<td>next rate if condition 1 is not satisfied;</td>
</tr>
<tr>
<td>capacity check RoT([j]) with “intracell-margin”;</td>
</tr>
<tr>
<td>next rate if condition 2 is not satisfied;</td>
</tr>
<tr>
<td>capacity check for all other cell RoT([\text{all cells}]) with “intercell-margin”;</td>
</tr>
<tr>
<td>next rate if condition 2 is not satisfied;</td>
</tr>
<tr>
<td>update LM([i, \text{all cells}]);</td>
</tr>
<tr>
<td>end</td>
</tr>
<tr>
<td>end</td>
</tr>
<tr>
<td>remove user ([i]) from sorted users of cell ([j])</td>
</tr>
<tr>
<td>end</td>
</tr>
<tr>
<td>end</td>
</tr>
</tbody>
</table>
Chapter 6. Load Matrix Scheduling

The first step is initialization where all the LM elements are set to zero and also users in each cell are sorted according to a priority function, such as (6.16).

The LM allocation process simultaneously increases allocated resources in each cell to avoid interference imbalance amongst the cells. The process consists of a number of assignment rounds equal to the maximum number of users per cell e.g. 10 rounds for 10 users per cell as stated in Table 6.3. In each round, the LM assigns rates to the highest priority user in each cell, updates LM elements and performs capacity checking. Capacity Check (CC) function calculates RoT as in (6.12) and compares with \( \text{RoT}_{\text{target}} \) making sure (6.9) is always satisfied. Passing this “check” means assigned rates are valid and will not cause interference outage. If CC fails, scheduler attempts the next available rate and continues until CC is satisfied. Then the user is considered scheduled and will be removed from the user priority list of its serving cell. The scheduling process continues until all the users are processed.

In the LM scheduling process, the CC function is especially important and plays a major role in the overall system performance. In particular, a margin concept rather than a fixed threshold for \( \text{RoT}_{\text{target}} \) is introduced. This is to minimize the fluctuation behaviour in RoT performance as observed in conventional scheduling (see section 5.2). The CC operates on this small margin around \( \text{RoT}_{\text{target}} \) instead of a fixed \( \text{RoT}_{\text{target}} \) threshold. Two independent margin variables called intercell margin and intracell margin are specified for handling intercell and intracell interference respectively and assist better decision during the CC process. The intracell margin is a region set around the \( \text{RoT}_{\text{target}} \) where a serving cell’s user loading should not exceed. Similarly, intercell margin is another region specified around \( \text{RoT}_{\text{target}} \) which limits variations of overall RoT caused by a user from other cells. The intercell and intracell margins can be equal or different resulting in different performances. It will be shown in the following section that maintaining RoT in a small margin around \( \text{RoT}_{\text{target}} \) instead of using a single \( \text{RoT}_{\text{target}} \) threshold will result generally in a much improved interference outage performance and higher resource utilization.

The concept of Load Matrix technique is generic and it is not restricted to any specific air interface technology and can be used in conjunction with any adaptive technique and priority function. \( \text{RoT}_{\text{target}} \) defines the maximum cell capacity even though the instantaneous capacity in a cell is not fixed. In other words, the CC function explained above regards a cell as fully loaded only when its estimated RoT reaches \( \text{RoT}_{\text{target}} \). That means cell capacity is not a direct function of the transmission rates being assigned. The LM technique operates on RoT, channel gains and the specified constraints such as user power and available rates. These parameters are common to all mobile cellular systems. However, their calculations are different for different air interface technologies, and dependant on the network architecture. Here for simulation purposes and in order to have a platform to prove the concept and to demonstrate the performance of the Load
Matrix, HSUPA system is chosen. Besides, HSUPA and HSDPA were the first standards which seriously address the importance of the scheduling problem.

6.3 Simulation Results

To evaluate the performance of LM scheduling, extensive system level simulations have been carried out using HSUPA [3] platform. The results are compared with the scheduler in [8] that is also used as a benchmark for comparison in [3]. Same simulation parameters used for both the benchmark and the LM technique as shown in Table 6.3. For simplicity, Hybrid Automatic Repeat reQuest (HARQ) is not considered here in neither of these algorithms.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>system layout</strong></td>
<td>Hexagonal grid, omni sites, 3 tiers (19 base stations) wrap around</td>
</tr>
<tr>
<td>Number of users</td>
<td>190 (10 users per cell)</td>
</tr>
<tr>
<td>Cell radius R</td>
<td>1.8 km</td>
</tr>
<tr>
<td>Path loss model</td>
<td>$L = 128.1 + 37.6 \log_{10}(R)$</td>
</tr>
<tr>
<td>Channel</td>
<td>AWGN + shadowing</td>
</tr>
<tr>
<td>Std. deviation of slow fading</td>
<td>8.0 dB</td>
</tr>
<tr>
<td>Correlation distance of slow fading</td>
<td>50 m</td>
</tr>
<tr>
<td>BS antenna gain plus Cable Loss</td>
<td>14 dBi</td>
</tr>
<tr>
<td>Carrier frequency</td>
<td>2 GHz</td>
</tr>
<tr>
<td>Rx antenna</td>
<td>1</td>
</tr>
<tr>
<td>Tx antenna</td>
<td>1</td>
</tr>
<tr>
<td>User antenna gain</td>
<td>0 dBi</td>
</tr>
<tr>
<td>Maximum User EIRP</td>
<td>21 dBm</td>
</tr>
<tr>
<td>Maximum BS EIRP</td>
<td>24 dBm</td>
</tr>
<tr>
<td>CL Power Control</td>
<td>1 dB step size</td>
</tr>
<tr>
<td>Transmission rates (kbit/s)</td>
<td>8,16,32,64,128,256,384</td>
</tr>
<tr>
<td>TTI</td>
<td>10 ms</td>
</tr>
<tr>
<td>Scheduling period</td>
<td>Every 10 TTI</td>
</tr>
<tr>
<td>Traffic model</td>
<td>Full buffer</td>
</tr>
<tr>
<td>Number of UEs per cell</td>
<td>10</td>
</tr>
<tr>
<td>Simulation time</td>
<td>20 s</td>
</tr>
<tr>
<td>RoT target</td>
<td>5.23 dB (= 70% load factor)</td>
</tr>
<tr>
<td>Number of iterations (in Load Matrix technique)</td>
<td>No iteration</td>
</tr>
</tbody>
</table>

It should be emphasised that our main objective is to highlight the impact of other cell interference existing in both centralized and decentralized scheduling algorithms. Another important objective is to observe the performance of scheduling algorithms compared with the
upper-bound limit rather than comparison between different algorithms because comparison with the upper-bound limit is a better indication of scheduling algorithm efficiency. Here, the upper-bound limit on the interference outage performance can be defined as a "step function" in CDF of RoT (as represented by target RoT in Figure 6-3, Figure 6-5 and Figure 6-7). Interference outage performance directly affects all other performance measures like throughput and packet delay. The comparison with the benchmark algorithm (Qualcomm, see section 2.3) is provided here as an example to show the effectiveness of the LM scheduling compared with a conventional scheduling algorithm used in [3] as reference. General comparison between centralized and decentralized scheduling is not in the scope as it is already available in literature e.g. in [19].

The system level simulator models 19 omni-directional cell structure with 10 users per cell randomly and uniformly distributed. The resource allocation performance is measured in terms of interference outage probability, averaged cell throughput and packet delay.

The simulation results provided here have two different objectives. The first objective is to show the impact of the margin concept (both intercell and intracell) on the interference outage performance. These are shown in Figure 6-2 and Figure 6-3 for intercell margin, and in Figure 6-4 and Figure 6-5 for intracell margin. The second objective is to illustrate the performance of the LM (based on the best margin setup) compared with the benchmark algorithm and the upper-bound limit in terms of interference outage.

![PDF of RoT](image)

*Figure 6-2: PDF of RoT (inter-cell margin effect)*
Figure 6-3: CDF of RoT (inter-cell margin effect)

Figure 6-2 and Figure 6-3 (in which no intracell margin is considered), show the Probability Density Function (PDF) and Cumulative Density Function (CDF) of RoT respectively for benchmark algorithm and three LM cases. The LM case “no hysteresis” represents the case with no intercell margin while the other two cases represent intercell margins of +1% and +2%. These figures demonstrate considerable improvement in interference outage performance compared with the benchmark algorithm and a close performance to the upper-bound limit (step function) indicating the sensitivity of performance to intercell margin.

Same behaviour can be observed in Figure 6-4 and Figure 6-5 where intracell margin is used instead of intercell.

It is interesting to note that intercell and intracell margins have positive and negative values with respect to $RoT_{target}$. This is due to the fact that under any conditions, the intracell interference is more dominant than the intercell interference. Therefore the LM capacity check has to be more strict with own cell users contribution to cell loading by not permitting RoT to exceed the $RoT_{target}$. However, it can show more flexibility in accommodating the other cell interference allowing their contribution to RoT slightly exceeding the $RoT_{target}$ but within the specified intercell margin.
Chapter 6. Load Matrix Scheduling

Figure 6-4: PDF of RoT (intra-cell margin effect)

Figure 6-5: CDF of RoT (intra-cell margin effect)
In order to find an appropriate set of intercell and intracell margin, we have carried out simulations for 25 combinations of \textit{intercell margin}=[0, 0.5\%, 1\%, 1.5\%, 2\%] and \textit{intracell margin}=[0, -0.5\%, -1\%, -1.5\%, -2\%]. Following performance figures represent the interference outage performance of the best three combinations:

- [0\% intracell margin, +1.5\% intercell margin]
- [-0.5\% intracell margin, +0.5\% intercell margin]
- [-1.5\% intracell margin, +0.5\% intercell margin]

![PDF of RoT](image)

\textbf{Figure 6-6: PDF of RoT (best three combinations)}

It is evident from the results that LM scheduling technique outperforms the benchmark algorithm in all cases, with a very close performance to the upper-bound limit. As stated before, our second objective in these simulations was to illustrate the performance of the LM (based on the best margin setup) compared with the benchmark algorithm and the upper-bound limit. Figure 6-6 and Figure 6-7 illustrate the performance in terms of interference outage. Service throughput versus distance from cell centre is depicted in Figure 6-8 and finally, PDF and CDF of Packet delay performances are illustrated in Figure 6-9 and Figure 6-10 respectively.

Figure 6-8 shows the service throughput versus distance from a base station for LM scheduler and the benchmark algorithm. It can be observed that in terms of cell throughput performance, the LM
outperforms the benchmark algorithm on average by more than 30%. The spikes in the cell throughput are due to the limited number of users generating transmit data.

Figure 6-7: CDF of RoT (best three combinations)

Figure 6-8: Average service throughput versus distance
Figure 6-9 and Figure 6-10 show the packet delay performance for LM scheduler ([-1.5% intracell margin, +0.5% intercell margin]) versus the benchmark algorithm. This again demonstrates another performance improvement. It can be seen from Figure 6-10 that with LM scheduling technique, 95% of the packets experience delay of less than 40 TTI compared with 200 TTI experienced by the benchmark algorithm. It is worth noting that all the LM results presented here are achieved with no iteration, and still resulting in negligible interference outage as can be observed comparing with the upper-bound limit.

![Packet delay histogram](image)

Figure 6-9: Histogram of Packet delay

Finally, it will be very interesting to recall the RoT fluctuation issue discussed in section 5.2 and to see where we are on that after implementing LM.

Figure 5-2 shows typical RoT fluctuation in a cell due to intercell interference using a conventional scheduler in a normal condition. In one scheduling instant, RoT of a cell dramatically increases to well above the threshold and in the following scheduling instant drops to well below the threshold. This is due to lack of information about neighbouring cells and hence negative impact in terms of interference outage which in turn increases the probability of packet errors. Figure 6-11 shows the RoT fluctuation in time in the exactly same cell as of that in Figure 5-2 in exactly same time period after implementing Load Matrix. It is not surprising to see that the fluctuation is almost none existent. Negligible variation around RoT_{target} is due to the hysteresis margin we used and it is favourable as explained already.
Figure 6-10: CDF of packet delay

Figure 6-11: RoT fluctuation with LM scheduler in the same cell/time as shown in Figure 5-2
6.4 LM - Priority functions combination

Priority functions are used to rank users in the scheduling process and make a balance between cell throughput and fairness. Commonly used priority functions in conventional scheduling are Round Robin (RR), SINR (also called MaxC/I), and Proportional Fair (PF) (see e.g. [3][7][9]). The Round Robin tries to maximize fairness amongst users regardless of a user channel condition and therefore results in poor throughput performance. Max C/I on the other hand, ranks users in terms of their channel quality and aiming for maximizing cell throughput at the expense of fairness. Proportional Fair performs better than Round Robin in terms of throughput and better than MaxC/I in terms of fairness.

As stated before, Load Matrix technique provides a generic solution for resource scheduling that does not preclude any priority function and can be combined with any of them. However, priority function has major impact on overall system performance for any scheduling technique including the LM. In this section we examine the possibility to combine LM with conventional priority functions aiming to benefit from efficient interference control mechanism provided by LM.

Usually, scheduling algorithms deploy different priority functions due to different objectives. Their performance depends also on the deployed system and the environment characteristics. Some algorithms, for instance, aim for fairness in resources given to the user whereas others are more focused on generating higher throughput. Round Robin (RR), Max C/I and Proportional Fair (PF) are considered here as the basis for analysis. RR is a fair and simple algorithm. Resources are allocated to users in a cyclic order offering fair resource sharing among them. However the property of not considering the radio channel condition produces very low throughput. On the contrast to RR, Max C/I is based on the channel conditions by allocating the available resources to the user with the best channel quality in terms of Signal to Interference ratio, and therefore increases the total system throughput. As a result, the users close to base station are more likely to have always better channel condition and therefore consume the resources. Max C/I increases the cell capacity but suffers from poor fairness.

PF increases the influence of previous transmission rates and allows trade-off between fairness and throughput. However, it tends to always select users with limited fading variation [31].

Recently proposed Score-Based (SB) scheduling [39] analyses the user's traffic performance and allocates a transmission rate according to the score measured. It provides fairness according to rate statistics and increases robustness to the channel condition. While in PF the prioritization of transmission rate is based on own average throughput, SB takes advantage of rate statistics but not necessarily the transmission rate itself.
On the contrary to traditional schedulers, Load Matrix takes the intercell interference information into account in order to avoid RoT outage. LM uses a database containing the load contribution of all active users in the network.

The key point of these LM-enhanced algorithms is to benefit from efficient interference control mechanism provided by LM. The main difference between the conventional and the LM-enhanced schedulers is that the probability of the RoT exceeding its target has been significantly reduced by introducing LM [40]. Regarding average cell throughput, Max C/I has the best throughput performance which is around 30% more than PF and about 70% more than RR. By introducing Load Matrix, the average cell throughput of Max C/I has increased by 15% while that of PF has slightly decreased due to the fact that it achieved more fairness as a trade-off [40].

In developing LM technique, we also introduced a new priority called Global Proportional Priority (GPP) (see section 6.2). The principle of GPP as defined in (6.16) is to take into account the intercell impact by considering interference contribution of the user to other cells. In comparison to conventional Max C/I, if a user has good channel to more than one base station, it will not be given the rate that would have been given in conventional Max C/I.

We observed in section 6.3 that Load Matrix with GPP benefits from 30% improvement in overall throughput over the benchmark PF algorithm (as in [3]). Also 95% of the packets experience delay of less than 40 TTI compared with 200 TTI in the benchmark PF.

In [40], we introduced another priority function to Load Matrix called Global Proportional Fair (GPF). The key property of GPF is to exploit Load Matrix in order to minimize the interference generated in a cell towards its neighbours while enhancing its fairness performance to users in the network:

\[
PRIORITY_i = \frac{G_{ij}}{\sum_{k \neq j} G_{ik}} \cdot \frac{1}{r_k}
\]

\[
r_k = \frac{1}{W} \sum_{w=1}^{W} r_{kw}
\]

where \(r_k\) is the average user \(k\) transmission rate over time window of \(W\). Following, the performance of GPF in terms of throughput, fairness and also range dependency are presented in details and compared with other algorithms.

Figure 6-12 illustrates the basic strength of LM regardless of the priority function used and that is the capability to control and maintain intercell interference. Similar to PF scheduler, the performance of GPF algorithm depends on the size of its averaging window \(W\) in (6.18). The window size \(W\) used for GPF in this figure is 10.
From interference outage point of view, GPF like any other LM-enhanced scheduler has strict control over interference generated and RoT has been maintained below the target regardless of the $W$ size.

Figure 6-13 shows 15% less average cell throughput for GPF compared with LM+Max C/I. This is due to the trade-off for its fairness performance outperforming conventional Max C/I.
significantly. That means GPF is capable of combining performance benefits of the two, i.e. the throughput advantage of conventional Max C/I and fairness advantage of PF. Not surprising to see average cell throughput in GPF is much higher than RR or PF as shown in Figure 6-13.

The window size $W=10$ is selected for GPF as an example to examine the performance. The size of $W$ can swing GPF performance between throughput and fairness. Better fairness can be achieved by increasing $W$ while better throughput can be produced by decreasing it.

Figure 6-14 illustrates the impact of averaging window size $W$ on the throughput versus cell range. A set of GPF performance is presented e.g. GPF(10) being GPF$|_{W=10}$. It can be observed that the higher goes $W$, the better becomes the cell-edge throughput and range fairness (although it comes at a price of overall throughput). It is therefore very important to make the right balance between throughput and fairness. It is also important to provide fair chance of transmission resources with respect to the user location in the cell.

![Throughput over Distance graph]

**Figure 6-14: Throughput over distance - effect of $W$ size in GPF**

Finally, Figure 6-15 compares the throughput versus range performance for Max C/I, PF and GPF ($W=20$). As expected, one can see that GPF outperforms Max C/I in terms of fairness over the range, most importantly at cell-edge, and provides higher throughput than PF over the whole cell range.
Chapter 6. Load Matrix Scheduling

6.5 Summary

In this chapter, first we described how the proposed Load Matrix scheduling technique can facilitate joint management of interference within and between cells for allocation of radio resources. Simulation results show significant improvement in RoT performance, resource utilization and overall network performance. Using the LM technique, average cell throughput can be increased on average by 30% compared to benchmark scheduling algorithms such as qualcomm [8]. Also 95% of the packets experience delay of less than 40 TTI compared with 200 TTI in the benchmark. Results also show that maintaining cell interference within a margin instead of a hard RoTtarget can significantly improve RoT performance and hence, resource utilization.

Second, the combination of LM with conventional priority functions presented and the improvements assessed. We introduced GPP priority function as adaptation to MaxC/I and GPF as adaptation to PF in order to take into account the intercell impact of channel gains. Regarding average cell throughput, Max C/I has the best throughput performance which is around 30% more than PF and about 70% more than RR. By introducing Load Matrix, the average cell throughput of Max C/I has increased by 15% while that of PF has slightly decreased due to the fact that it achieved more fairness as a trade-off.
So far, we have not considered practical impairments in LM performance. For instance, these impairments can occur as the result of channel estimation error, channel information delay causing inaccuracy, and restriction of information to neighbouring cells only. Our discussion in previous chapters were based on perfect knowledge of channel information and without considering additional delay (in excess of one TTI already considered) in order to explore the upper-bound limits of system capacity in the Load Matrix technique.

In the next chapter, we study the effects of data impairments in the Load Matrix technique and its consequences on the system performance in terms of cell throughput degradation, Interference outage and delay. It is interesting from practicality point of view to assess the implementation issues such as signalling overhead, sensitivity of Load Matrix to channel measurement errors and signalling delay.
Chapter 7

7 Assessment on Load Matrix impairments

7.1 Introduction

The Load Matrix and its evaluation results discussed so far are based on perfect knowledge of channel information from active users without additional delay. In this chapter, we study the effects of data impairments in the Load Matrix and its consequences on the system performance namely cell throughput and interference outage. The aim of Load Matrix implementation as presented and discussed in chapter 6 based on perfect knowledge of channel information and without considering additional delay (in excess of one TTI delay already considered) was to explore the upper-bound limits of system capacity in the Load Matrix technique. It is interesting however from practicality point of view to further study the implementation issues such as signalling overhead, sensitivity of Load Matrix to channel measurement errors and delay.

In this chapter, first a closer look at Load Matrix database is provided and load element projection is discussed. It will be shown in section 7.2 that Load Matrix is in fact a sparse matrix in nature and therefore the number of load elements which are significant enough to be considered for this technique and eventually the signalling overhead is actually not large.

Another practical impairment studied in this chapter is the impact of channel estimation and/or measurement error on the LM performance. The effect of channel errors is investigated in section 7.3 and system performance is evaluated and compared under different error conditions. Given the fact that LM is a sparse matrix, in section 7.4 we investigate the effect of LM scheduling while being restricted to neighbouring cells only and extend the results of channel error to this study as well. Finally in section 7.5 the effect of additional delay in the channel information used by LM is presented and simulation results are discussed.
Chapter 7. Assessment on Load Matrix impairments

7.2 Load Matrix projection

Recalling the definition of Load matrix from chapter 6 as a database containing the load factors of all active users in the network, in a network of M users and N cells, it will become a $M \times N$ matrix with elements defined as

$$\begin{align*}
LM_{i,j} &= \frac{P_i G_{ij}}{N' + \sum_{i=1}^{M} P_i G_{ij}} , i \in \{1, \ldots, M\} , j \in \{1, \ldots, N\} \\
\end{align*}$$

where $N'$ is the thermal noise power at receiving base station (BS), $P_i$ represents the transmission power of User Terminal (UT) $i$ and $G_{ij}$ is the channel gain from user $i$ to BS $j$.

Each column in the load matrix represents an individual cell. Let us call it here Load Vector (LV) for that particular cell, as it contains the load participation from all active users in the network.

$$\begin{align*}
\overline{LV}_j &= [LM_{i,j}] , \forall i \in \{1, \ldots, M\} \\
\end{align*}$$

One can distinguish between elements of a load vector (which we call it Load Element generated by and associated to an individual user) based on the user's location in the network.

Figure 7-1 shows three different categories of load element relative to the location of its associated User Terminal (UT) and also with respect to a particular cell. These are Own cell elements (green), Neighbour elements (red) and Other elements (blue). For a particular LV$_j$, an Own cell element (as the name suggests) represents the load element participated by a UT located in the cell $j$. In other words, it will be seen and counted as intracell interference for other UTs in that cell. Neighbour elements represent the load elements participated by those users located in the neighbouring (adjacent) cells. Using familiar terms from interference management, it is actually the intercell interference generated by (and only by) the users in the first tier of cells around cell $j$. Finally, Other elements represent the intercell interference generated by users from second tier cells and beyond meaning all the rest of active users in the network. Figure 7-2 demonstrates a typical snapshot of Load Matrix taken when a conventional scheduling [8] is implemented in the system. The simulated network is constituted of 19 omni directional cells. In each cell there are 10 active users with full buffer. Users are randomly and uniformly distributed over the cell coverage area. Simulation parameters are same as in Table 6.3. It should be noted that, in this case we only use Load matrix database containing the load factors of all active users in the network and we have not used this information in the scheduling process. The green pixels
in Figure 7-2 represent the load elements generated by own cell users while the red and blue pixels represent intercell load from neighbour cells and other cells respectively.

\[
i = (j - 1) \cdot 10 + n, \forall n \in \{1, ..., 10\}
\]

The size of each pixel in Figure 7-2 is an indication of its load and is relative to the volume of load being generated meaning the bigger the pixel, the bigger the load element.

As expected, green pixels appear dominant in each cell. However, there are some significant red pixels as well which causes significant intercell interference. The blue pixels are not significant both in number and size. For better visibility, we only show the load elements bigger than one thousandth of RoT_{target} in Figure 7-2.

It appears from the snapshot projection that Load Matrix database is sparse i.e. the number of significant load elements outside the own cell is small.

Having said that, we should also remind that Figure 7-2 shows a handful of very strong intercell load elements (red pixels), some of them appear to be as strong as own cell elements.
Also interesting to note, is that blue pixels appear to be very small both in size and population. This means amongst intercell load elements, only some users from neighbour cells are generating significant participation to the load of their adjacent cell. These are important observations which help to build a better picture of load matrix and in our assessment on the implementation issues which we will discuss further in this chapter.

![Load Matrix projection](image)

Figure 7-2: A typical snapshot of Load Matrix projection in a cell for significant elements (bigger than RoT_arge/1000)

Let us have a closer look at Load Matrix elements by examining two LVs taken from Figure 7-2 representing cell 1 and cell 12. These two cells are chosen as they appear to be the best case and the worst case in terms of intercell load participation (see Figure 7-2) and also to highlight the observation about importance of neighbour cells. Cell 1 has strong red pixel showing significant intercell interference while cell 12 has neither a significant red nor blue pixel.

Figure 7-3 illustrates the load vector projection for cell 12. This is an example when most of the cell load is generated by its own users (in this case about 90%) and load participation from neighbours and other cells are negligible. This is also the case when conventional scheduling like [8] does well as there is not much intercell interference to worry about. It is, in a sense, an ideal situation for decentralised scheduling as every user are transmitting and yet the cell RoT is kept below the target 5.2 dB.
Chapter 7. Assessment on Load Matrix impairments

On the contrary to cell 12, cell 1 has significant intercell load contribution as illustrated in Figure 7-4. This is a case when most of the cell load is not generated by its own users but by intercell interference. About two thirds of the cell load here is due to interference coming from neighbour cells and about 8% from users even further away. This is an example when conventional scheduling like Qualcomm[8] comes short to maintain interference level and eventually the cell load. As a result, cell RoT surpass the target 5.2 dB. It is very interesting to note that cell capacity in this case has mostly consumed by those other than own cell users. As stated before, Figure 7-2 demonstrates a typical snapshot of Load Matrix taken when a conventional scheduling [8] is implemented in the system. That means we only have used Load matrix database containing the load factors of all active users in the network and we have not used this information in the scheduling process.

Now let us repeat the same scenario and examine Load Matrix projection in exactly same snapshot but when Load Matrix is used as the scheduler. The result is shown in Figure 7-5. This time, only green pixels appear as main load contributors in each cell. There is no significant red or blue pixel representing intercell interference.
Chapter 7. Assessment on Load Matrix impairments

Figure 7-4: Load Vector projection for cell no. 1 (worst case)

Figure 7-5: Load Matrix projection after implementing LM scheduler


## 7.3 Load Matrix with channel estimation/measurement error

The Load Matrix and its evaluation results presented in chapter 6 were based on perfect knowledge of channel information from active users. It is interesting however to further study the implementation issues such as sensitivity of Load Matrix to channel measurement errors. In this section we study the effects of inaccuracy in channel gains information used by Load Matrix and its consequences on the system performance namely cell throughput and Interference outage.

Recall Load elements defined in (7.1), let us replace it with

\[
LM_{i,j} = \frac{P_i(G_{i,j} + \delta_{i,j})}{N' + \sum_{i=1}^{M} P_i(G_{i,j} + \delta_{i,j})}, \quad i \in \{1, \ldots, M\}, \quad j \in \{1, \ldots, N\}
\]

(7.4)

Here, \(\delta_{i,j}\) is a normal random variable with mean equal to zero, added to the perfect channel gain \(G_{i,j}\) in order to represent error in channel estimation/measurement between user \(i\) and cell \(j\).

Figure 7-6 and Figure 7-7 show the Probability Density Function (PDF) and Cumulative Density Function (CDF) of RoT respectively for benchmark algorithm (qualcomm), Load Matrix scheduler with perfect channel information (no error) and three LM cases with channel error. For LM cases with channel measurement error, the difference is in Standard Deviation (STD) of \(\delta_{i,j}\) in (7.4) used in each case being 1 dB, 2 dB or 5 dB. First interesting observation is that even with 5 dB channel estimation/measurement error, LM scheduler performs far better than benchmark algorithm. Obviously, if the error excessively grows higher, the main advantage of LM being coordination across cells will be lost and performance drops to those of conventional schedulers like the benchmark.

Comparing LM cases with and without error, one can see from these figures that LM scheduling performance in terms of interference outage control is resilient to 1~2 dB channel error. This actually is an important conclusion that LM scheduling is robust enough against channel measurement/estimation error.
Chapter 7. Assessment on Load Matrix impairments

Figure 7-6: PDF of RoT in LM scheduling with channel error

Figure 7-7: CDF of RoT in LM scheduling with channel error
Now let us look at the impact of channel error in terms of system throughput and average packet delay. Service throughput versus distance from cell centre is depicted in Figure 7-8 and CDF of Packet delay performances is illustrated in Figure 7-9. It can be observed that in terms of cell throughput performance, the LM scheduler outperforms the benchmark algorithm significantly even with channel estimation/measurement error of 5 dB STD. Therefore in terms of throughput also, we can conclude LM is reliably robust against channel error. The spikes in the cell throughput are due to the limited number of users generating transmit data.

![Figure 7-8: Cell throughput versus distance with and without channel error](image)

This is also true in terms of packet delay performance, see Figure 7-9. It can be seen that with LM scheduling technique, between 92–97% of the packets experience delay of less than 50 TTI over the range of channel error STD up to 5 dB. This is significantly better compared with the benchmark algorithm (200 TTI experienced by the benchmark algorithm at 95%). Comparing LM cases with and without error, one can see from these figures that LM scheduling performance in terms of throughput and packet delay is also resilient to channel error of about 2 dB. Therefore, we can conclude that LM scheduling is robust enough against channel measurement/estimation error.
Chapter 7: Assessment on Load Matrix impairments

7.4 Load Matrix restricted to neighbour elements

In section 7.2, we observed that Load Matrix is a *sparse* matrix in nature. In other words, the population of load elements which are significant enough to be considered for LM scheduling technique and eventually the signalling overhead is not large. Recalling Figure 7-1, three different categories of load element identified (with respect to the location of its associated UT and also with respect to a particular cell) as *Own cell* elements (green), *Neighbour* elements (red) and *Other* elements (blue). *Neighbour* elements represent the load elements participated by those users located in the neighbouring (adjacent) cells. We observed a handful of strong *Neighbour* elements, some of them appear to be as strong as own cell elements. On the contrary, *other* elements (blue pixels) appear to be very small both in size and population. This means amongst intercell load elements, only some users from neighbour cells are generating significant participation to the load of their adjacent cells.

In this section, we examine this fact further by restricting the LM database to neighbouring cells only. For user $i$ and base station $j$, the following Load Matrix element definition is used to reflect this restriction.
Chapter 7. Assessment on Load Matrix impairments

\[ LM_{i,j} = \frac{P_i(G_{i,j} + \delta_{i,j})}{N^* \sum_{i=1}^{M} P_i(G_{i,j} + \delta_{i,j})} \text{, if } j \text{ is user } i \text{'s serving base station or neighbour} \quad (7.5) \]

\[ LM_{i,j} = 0 \text{, if } j \text{ is neither user } i \text{'s serving base station nor its neighbour} \quad (7.6) \]

Referring to Figure 7-2, this means we eliminate all the blue pixels in Load Matrix database and only consider red pixels during LM scheduling process. The aim is to verify the robustness of LM in the absence of intercell information from second tier cells by restriction of information to intercell interference from adjacent cells. From practicality point of view, this will result in significant reduction in the signalling overhead. Let us compare the system performance for the following different scheduling cases:

- Benchmark algorithm (Qualcomm)
- LM scheduler with perfect database (no error)
- LM scheduler with channel error as explained in previous section
- LM scheduler with database restricted to neighbour elements (neighbours only)
- LM scheduler with combination of restriction to neighbours and also channel error

Figure 7-10 and Figure 7-11 illustrate the Probability Density Function (PDF) and Cumulative Density Function (CDF) of RoT respectively, for benchmark algorithm (Qualcomm), Load Matrix scheduler with perfect channel information (no error), LM with 5 dB channel error, LM restricted to neighbours with and without channel error.

First, let start by comparing perfect LM (no error) case with LM case restricted to neighbours only. Figures clearly show that the difference is negligible, in fact hard to notice any difference. This fact leads to a fundamental conclusion: LM intercell interface control can be achieved satisfactorily by coordination only amongst adjacent cells. Furthermore, coordination across adjacent cells is not necessary for all the users but those who has significant load element in an adjacent cell.
Chapter 7. Assessment on Load Matrix impairments

Figure 7-10: PDF of RoT in LM scheduling with restriction to neighbours and with channel errors

Figure 7-11: CDF of RoT in LM scheduling with restriction to neighbours with channel errors
Second observation comes when channel error is applied, by comparing LM (5 dB error) case with LM case restricted to neighbours and with channel error (neighbours 5 dB error). Figure 7-10 and Figure 7-11 show considerable difference in interference outage performance. Basically, one can see that LM when restricted to neighbour cells, is more sensitive to channel error compared to when it is not restricted. Although we could tolerate up to 5 dB channel error in case of LM and still have good interference outage performance, in case of LM restricted to neighbours it becomes 1–2 dB. Therefore, we can rephrase our conclusion as following: LM intercell interface control can be achieved satisfactorily by coordination amongst adjacent cells using reliable channel estimation/measurement techniques with error margin less than 2 dB STD.

Now let us look at the throughput performance when LM is restricted to neighbours. Service throughput versus distance from cell centre is depicted in Figure 7-12. In the previous section, we already observed that LM scheduler outperforms the benchmark algorithm significantly even with channel estimation/measurement error of 5 dB STD. Here we can see that restricted LM also performs very closely to LM with 1–2 dB channel error case in terms of throughput. The spikes in the cell throughput are due to the limited number of users generating transmit data.

![Figure 7-12: Cell throughput versus distance with and without channel error](image-url)
Chapter 7. Assessment on Load Matrix impairments

Figure 7-13: CDF of packet delay in LM scheduling with restriction to neighbours with and without channel error

Figure 7-13 illustrates LM packet delay performance with and without restriction to neighbour elements. It can be seen again that the difference is negligible, in fact hard to notice any difference. With LM scheduling with and without restriction to neighbours, around 95% of the packets experience delay of less than 50 TTI. When we introduce 5 dB STD channel error into restricted LM (neighbours 5 dB error case), packet delay is less than 75 TTI at 95%. This is still far better than benchmark algorithm (200 TTI experienced by the benchmark algorithm at 95%).

By comparing LM performances with and without restriction to neighbour elements, one can see that LM scheduling performance in terms of interference outage, throughput and packet delay is resilient to this restriction. Therefore, we can say LM scheduling can be performed satisfactorily by coordination amongst adjacent cells only using reliable channel estimation/measurement techniques with error margin less than 2 dB STD.
7.5 Load Matrix with additional delay

The Load Matrix and its evaluation results presented in chapter 6 were based on perfect knowledge of channel information from all the active users without additional delay in order to explore the upper-bound limits of system capacity in the Load Matrix technique. In previous sections of this chapter, we investigated the effects of data impairments in the Load Matrix and its consequences on the system performance namely cell throughput, Interference outage and packet delay. It was shown that Load Matrix is in fact a very sparse matrix and therefore the volume of channel information required for this technique and eventually the signalling overhead is actually not large. Another practical impairment studied was channel estimation/measurement impairments. Simulation results presented show that LM scheduling is reliably robust against channel impairments up to 2 dB STD error. Given the fact that LM database is a sparse matrix, we then studied the effect of LM scheduling while being restricted to neighbouring cells only and extend the results of channel error to this study as well. We observed that LM scheduling can keep its performance well above conventional scheduling even when restricted to coordination amongst adjacent cells only. Dominant intercell interferers proved to be from neighbour cells not second tier cells or beyond.

In this section, the effect of additional delay in the channel information used by LM is studied and simulation results are discussed. The LM results discussed and presented so far were all based on one TTI information delay in LM database and without considering additional delay (in excess of one TTI already considered) in order to explore the upper-bound limits of system capacity in the Load Matrix technique. That means at each scheduling instant, LM scheduler uses channel information updated at previous TTI. At scheduling instant $T_i$, for user $i$ and base station $j$, the Load Matrix element $LM_{ij}$ with additional delay of $\beta \cdot TTI$ (on top of one TTI delay already considered in LM implementation) can be written as

$$LM_{ij}(T_i) = \frac{P_i \cdot G_{i,j}(t = T_i - (\beta + 1) \cdot TTI)}{\sum_{m=1}^{N} P_i \cdot G_{m,j}(t = T_i - (\beta + 1) \cdot TTI)}$$  

(7.7)

Following, we introduce additional delay in channel information and examine LM performance in this case and assess the degradation caused. LM scheduling with additional 1 and 2 TTIs (i.e. $\beta = 1, 2$) will be presented and compared with benchmark algorithm (qualcomm) as well as LM with no additional delay.

Figure 7-14 and Figure 7-15 illustrate the Probability Density Function (PDF) and Cumulative Density Function (CDF) of RoT respectively for benchmark algorithm, Load Matrix scheduler
Chapter 7. Assessment on Load Matrix impairments

with perfect channel information (no error), and two LM cases with additional 1 and 2 TTI delay implemented. Comparing perfect LM (no error) case with LM cases with additional delay, figures show some degree of performance degradation. In case of 1 TTI additional delay however, the difference is negligible. Comparing with benchmark algorithm, both cases of LM with additional delay still perform far better in terms of interference outage performance.

It is therefore fair to say that LM intercell interface control is satisfactory with up to 2TTI delay in channel information prior to scheduling instant.

![PDF of RoT]

Figure 7-14: PDF of RoT in LM scheduling with additional delay

Now let us look at the throughput performance when additional delay in channel information is implemented in LM scheduling. Service throughput versus distance from cell centre for LM with additional 1TTI and 2 TTI delay is illustrated in Figure 7-16. It can be seen that introduction of additional delay of up to 2TTI in channel information in LM database does not have a noticeable degradation impact on the throughput performance. LM scheduler with additional delay still outperforms the benchmark algorithm significantly. The spikes in the cell throughput are due to the limited number of users generating transmit data.
Chapter 7. Assessment on Load Matrix impairments

Figure 7-15: CDF of RoT in LM scheduling with additional delay

Figure 7-16: Cell throughput versus distance for LM scheduling with additional delay
Figure 7-17 illustrates LM packet delay performance with and without additional delay in channel information. It can be seen again that the difference is negligible, if any. With LM scheduling with and without additional delay of up to 2TTI, around 95% of the packets experience delay of less than 50 TTI. This is far better than benchmark algorithm (200 TTI experienced by the benchmark algorithm at 95%).

![CDF of packet delay](image)

**Figure 7-17: CDF of packet delay in LM scheduling with additional delay**

### 7.6 Summary

In this chapter, we studied the effects of data impairments in the Load Matrix and its consequences on the system performance namely cell throughput and Interference outage. First, a closer look at Load Matrix database is provided and load element projection is discussed. It was shown that Load Matrix is in fact a *sparse* matrix in nature and therefore the number of load elements which are significant enough to be considered for this technique and eventually the signalling overhead is actually not large. Another practical impairment studied in this chapter is the impact of channel estimation and/or measurement error on the LM performance. Given the fact that LM is a sparse matrix, we illustrated the effect of LM scheduling while being restricted to adjacent cells only and extended the results of channel error to this study as well. Finally the effect of additional delay in the channel information used by LM is presented and simulation results were discussed.
Overall, by comparing LM performances with and without impairments studied in this chapter, following conclusions and assessments can be drawn:

- LM scheduling performance in terms of interference outage, throughput and packet delay is resilient to channel estimation/measurement error of up to 5 dB STD.

- LM database can be restricted to neighbour elements. That means LM scheduling can be performed satisfactorily by coordination amongst adjacent cells only, when reliable channel estimation/measurement techniques with error margin less than 2 dB STD is used.

- LM scheduling performance in terms of interference outage, throughput and packet delay is resilient to channel information delay of up to 2 TTIs.
Chapter 8

8 Stability of Load Matrix

An important aspect that must be considered in developing new techniques for Radio Resource Management is stability, longevity and uniformity of performance, in other words; consistency of performance across the network and over time.

In this thesis so far, we introduced and discussed Load Matrix as the way forward for future scheduling and multi-cell interference control. Detailed evaluation results were presented and discussed with and without perfect knowledge of channel information, restriction of information to neighbour cells and also additional delay. First, we introduced LM in chapter 6 based on perfect knowledge of channel information and without considering additional delay in order to explore the upper-bound limits of system capacity in the Load Matrix technique. Then in chapter 7, we extended our vision towards practicality point of view and further study the implementation issues such as signalling overhead, sensitivity of Load Matrix to channel measurement errors and signalling delay. It was shown that Load Matrix is in fact a very sparse matrix and therefore the volume of significant data required for this technique and eventually the signalling overhead is actually not large. Overall, by comparing LM performances with and without impairments studied in chapter 7, we observed that LM scheduling performance in terms of interference outage, throughput and packet delay is resilient enough to reasonable level of channel estimation/measurement error as well as signalling delay. As a result we conclude that LM database can be restricted to neighbour elements only. In other words, LM scheduling can be performed satisfactorily by coordination amongst adjacent cells, when robust and reliable channel estimation/measurement techniques are used. In this chapter and as the closing point on LM performance assessments, we study the Stability of Load Matrix technique and compare it with conventional scheduling in terms of consistency on interference control and throughput across the network (uniformity) and over time (longevity).
8.1 Stability in interference control

Let start with interference outage performance and have a closer look at RoT stability and longevity. In chapter 5, the main shortcoming of conventional decentralized scheduling was explained. In decentralized scheduling, each base station assigns radio resources (i.e. transmission rate and time) to its users until the estimated RoT reaches a predefined target value, $\text{RoT}_{\text{target}}$. $\text{RoT}_{\text{target}}$ is usually a fixed target value set by the network controller to maintain the uplink interference level [3]. It is true that Intracell interference is generated by own cell users and it is manageable by serving base station. However the main shortcoming for conventional decentralized scheduling in general becomes more evident in a multi-cell scenario where a considerable proportion of RoT is intercell interference and base station has little knowledge about and control upon. Figure 5-2 illustrates RoT fluctuation in this case due to intercell interference. Obviously, there is no stability across the network when RoT of each cell fluctuates in time. In Figure 6-11, we observed how LM can maintain RoT fluctuation in a very small margin around $\text{RoT}_{\text{target}}$.

Now let us examine RoT stability over time for benchmark scheduler (qualcomm) for the best cell case and worst cell case and compare the same with LM. We pick two cells from benchmark scheduler scenario which has least RoT fluctuation (best cell) and most RoT fluctuation (worst cell) and call them cell “1” and “2” respectively. Note the term “best” and “worst” here are used only from interference point of view. Figure 8-1 shows cell RoT over time for these two cells under benchmark scheduler and LM scheduler. For the benchmark scheduler, Cell 1 shows reasonably stable RoT over time (4.5 dB ~6 dB) while Cell 2 suffers from very large RoT fluctuation range between 2 dB and 9 dB. This is despite the fact that $\text{RoT}_{\text{target}}$ is set to 5.23 dB for both cells. It is a very different story for the very same cells when LM scheduler is implemented. Basically there is no “best” and “worst” cell in this case as LM manages RoT very close to $\text{RoT}_{\text{target}}$ for both cells at all time. For both cells, RoT variation is within 4.5 dB to 5.5 dB range.

It will be interesting to see how RoT behaviour is across the whole network. Figure 8-2 illustrates the 3D plot of RoT when the benchmark scheduler is implemented. In fact, Figure 8-1 is a vertical slice taken from Figure 8-2. The colour code used in this plot is correspondent to the magnitude of data. It is evident that RoT is completely unstable with a wide range of variation both in time and across cells.
Chapter 8. Stability of Load Matrix

Cell RoT stability over time

Figure 8-1: Cell RoT Stability comparison (best cell and worst cell)

Figure 8-2: Qualcomm 3D RoT plot
Now let examine RoT stability for LM scheduler. In Figure 8-1 we saw cell RoT over time for Cell 1 and Cell 2 that perform so differently under the benchmark scheduler. As mentioned, there is nothing as “best” and “worst” cell in LM case as it maintains RoT very close to RoT$_{target}$ for all cells at all time.

Figure 8-3 illustrates the 3D plot of RoT when Load Matrix is implemented. The same colour code as in Figure 8-2 is used in this plot.

It is clear that RoT under Load Matrix is completely stable and interference is consistently maintained within a narrow range of variation both in time and across the whole network.

Figure 8-3: Load Matrix 3D RoT plot
8.2 Stability in Throughput

In the previous section, we demonstrated the stability achieved on RoT across the entire network via interference control enforced by Load Matrix. It was shown that LM not only stabilizes RoT in a cell over time but also stabilizes RoT across cells (uniformity) and maintains it close to RoT_{target} to make the most of available radio resources. However, the ultimate goal is to stabilize system throughput. Theoretically, it is hard to draw an accurate one-to-one relationship between RoT and throughput.

In this section, we will show this fact that throughput can and will also be stabilized as direct result of interference control by implementing Load Matrix. Again, let start with same two cells from benchmark scheduler scenario (i.e. best cell and worst cell) as in section 8.1. Figure 8-4 shows cell throughput over time for these two cells under benchmark scheduler and LM scheduler. For the benchmark scheduler, even though Cell 1 had reasonably stable RoT over time, in terms of throughput it is not the case. Obviously, Cell 2 which suffers from very large RoT fluctuation has inevitably higher throughput variation. The main difference is in average cell throughput which is higher in Cell 1 by about 30–40%.

As in previous section, when LM scheduler is implemented there is no “best” and “worst” cell as LM stabilizes throughput for both, see Figure 8-4. The difference is that average cell throughput for Cell 1 is higher than Cell 2 by about 10%. Comparing benchmark cases with LM, one can see that Load Matrix has not only increased cell throughput (by 40% for Cell 1 and 50% for Cell 2) but also stabilizes that for both cells.

Figure 8-5 illustrates the 3D plot of throughput when the benchmark scheduler is implemented. Again, the colour code used in this plot is correspondent to the magnitude of data. As expected, throughput is completely unstable with a wide range of variation both in time and across cells. It is fair to say the Benchmark scheduler has failed to provide a sustainable cell throughput. It shows that unstable throughput performance observed in Figure 8-4 is actually spread over all cells.
Chapter 8. Stability of Load Matrix

Cell throughput stability over time (best cell, worst cell)

Figure 8-4: Cell Throughput Stability (best cell and worst cell)

Figure 8-5: Qualcomm 3D Throughput Stability plot
It will be interesting to see the same plot when Load Matrix is implemented. In Figure 8-4 we observed that cell throughput over time for both Cell 1 and Cell 2 are unstable and less efficient when the benchmark scheduler is implemented. We also observe that the concept of "best" and "worst" cell is none-existent in LM case as it maintains cell throughput for both cells.

Figure 8-6 illustrates the 3D plot of cell throughput when Load Matrix is implemented. The same colour code as in Figure 8-5 is used in this plot. It proves the fact that stable RoT performance in Figure 8-3 can lead to a sustained stable and uniform throughput performance across all cells over time.

8.3 Summary

An important aspect that must be considered in developing new techniques for Radio Resource Management is stability of performance. In other words, RRM techniques must not only improve the performance in terms of interference outage, throughput and delay, but also provide consistency of performance across the network (uniformity) and over time (longevity). In this chapter, we illustrated the fact that both cell RoT and cell throughput can and will be stabilized as direct result of interference control by implementing Load Matrix.
Chapter 8. Stability of Load Matrix

First, we examined RoT stability over time for benchmark scheduler (Qualcomm) for the best cell case (Cell 1) and worst cell case (Cell 2) and compare the same with LM. For the benchmark scheduler, Cell 1 shows reasonably stable RoT over time (4.5 dB ~ 6 dB) while Cell 2 suffers from very large RoT fluctuation range between 2 dB and 9 dB. When LM scheduler is implemented, basically there is no “best” and “worst” cell as LM manages RoT very close to target level for all cells at all time. For both cells, RoT variation is within 4.5 dB to 5.5 dB range.

Second, cell throughput over time for these two cells under benchmark scheduler and LM scheduler are examined. For the benchmark scheduler, even though Cell 1 had reasonably stable RoT over time, in terms of throughput it is not the case. Obviously, Cell 2 which suffers from very large RoT fluctuation has inevitably higher throughput variation. The main difference is in average cell throughput which is higher in Cell 1 by about 30–40%. Again, when LM scheduler is implemented there is no “best” and “worst” cell as LM stabilizes throughput for both. The only difference is that average cell throughput for Cell 1 is higher than Cell 2 by about 10%.

Comparing benchmark cases with LM, one can see that Load Matrix has not only increased cell throughput (by 40% for Cell 1 and 50% for Cell 2) but also stabilizes that for both cells.

Overall, we can conclude that Load Matrix provides uniformity and longevity in interference outage and throughput performance, consistently maintains it within a narrow range of variation both in time and across the whole network.
Chapter 9

9 Conclusion and Future Work

9.1 Conclusion

Future ubiquitous mobile communications require new RRM schemes to guarantee high and stable capacity and quality of service across the whole network. These conflicting requirements can be achieved to large extent through new research approaches on system level by appropriately exploiting the dynamics of multi-cell system and the system side information. This side information comes from multi-user and multi-cell characteristics of a cellular network and when utilised properly, would lead to several order of magnitude in capacity increment compared to that obtained purely from advanced modulation and coding schemes at the link level. This sets a new trend in research direction for future mobile communications. To date on the system level, conventional techniques have approached RRM mainly focusing on resources within a cell and to large extent ignoring effects of multi-cell.

In the first part of this thesis, we analysed conventional Uplink scheduling and identified the main causes for resource utilization inefficiency to be due to lack of consideration of effect of multi-cell interference in the process. In a decentralized scheduling scheme each base station assigns radio resources to its associated users on a priority basis until the estimated loading (or RoT in case of CDMA) reaches its pre-defined target. The main advantage of decentralized scheduling algorithms as opposed to centralized ones is less signalling overheads. We further exploited and extended the advantage of decentralized scheduling by developing two distributed scheduling algorithms (UniS Algo-1, Algo-2) in which a user terminal decides on the uplink transmission rate instead of a RNC or a base station. It is shown that the proposed distributed algorithms can improve performance of the decentralized scheduling in terms of throughput, packet delay and buffer occupancy. However, when the system loading was increased, the performance gain achieved by distributed algorithms over conventional scheduling was negligible. It was also observed that for those users close to cell centre, distributed scheduling algorithms achieved superior performance compared to conventional scheduling but not for cell edge users.
examining the behaviour of results, we identified an inherent conceptual shortcoming from conventional scheduling algorithms which also affected the performance of the distributed scheduling ones and prevented the system reaching its maximum capacity.

This led us to a second and main part of this thesis. A novel technique called Load Matrix (LM) was proposed which enables joint management of interference and scheduling within and between cells. The LM addresses and takes into consideration the influence of multi-cell interference on overall radio resource utilisation, setting a new direction for future research on resource scheduling in a multi-cell system. The main shortcoming in conventional scheduling was proved to be as follow; a significant proportion of RoT comes from other cell interference contributions which base station has little knowledge and control of. The vulnerability of conventional resource allocation and scheduling techniques to intercell interference results in interference fluctuations and hence capacity instability. Such interference fluctuation results in capacity wastage and excessive packet delay performance. Detailed look into this problem signified the importance of Intercell interference control as a key factor in future RRM designs. By formulating resource allocation problem in a cellular system, we proved that finding optimum resource allocation is in fact an NP-hard problem. The LM technique provided an efficient, practical and easy to implement solution and is considered as the way forward for scheduling specification for the future mobile broadband networks. Simulation results showed significant improvement in the resource utilization, overall network performance stability, longevity and uniformity. It was shown that by using the LM technique, average cell throughput can be increased as much as 30% compared to the conventional scheduling one used. Results also showed that maintaining cell interference within a margin instead of a hard target (RoT target) can significantly improve the performance and hence increase resource utilization. The Load Matrix technique provides an efficient resource allocation method through control and coordination of intercell interference. Extensive LM simulation results demonstrated its superiority, stability and robustness over conventional scheduling in terms of both averaged packet delay and cell throughput. The interference outage performance was also proved to be very stable, close to the upper-bound limit. By incorporating a new concept of separate and independent margins for intercell and intracell interferences into the LM, it was shown that better control over the two types of interference can be achieved resulting in high overall network performance. Through the LM technique, interference can be always kept close to a pre-determined target whilst average cell throughput can be increased by more than 30% compared with the benchmark scheduling algorithm.

From practicality of the LM point of view, implementation issues such as signalling overhead, sensitivity to channel measurement errors and additional delay were investigated and effects of impairments in the Load Matrix and its consequences on the system performance namely cell throughput and Interference outage were evaluated. The Load Matrix database projection showed
that the LM is in fact a sparse matrix in nature and therefore the number of load elements which are significant enough to be considered for this technique and eventually the signalling overhead are actually small. The effect of channel gain errors was also investigated and system performance was again evaluated and compared under different error conditions, demonstrating the LM robustness. Given the fact that LM is a sparse matrix, the performance of LM scheduling whilst being restricted to neighbouring (adjacent) cells were studied in conjunction with the results of the channel gain errors. Additionally, the effect of delays in the channel information on the performance of the LM was investigated. In summary, by comparing LM performances with and without impairments, following conclusions and assessments were observed:

- LM performance in terms of interference outage, throughput and packet delay was resilient to channel estimation/measurement error.
- LM scheduling performance in terms of interference outage, throughput and packet delay was resilient to Channel State Information (CSI) delay.
- LM database can be restricted to neighbour elements thus reducing signalling overhead. LM technique can be performed satisfactorily by coordination amongst adjacent cells only, provided robust channel estimation/measurement technique is used.

Finally, an important and interesting observation to make on LM technique was its stability of performance across the entire network (uniformity) and over time (longevity). It is shown that LM not only stabilizes RoT in a cell over time but also stabilizes RoT across cells uniformly. This is in contrast to conventional scheduling which throughput was completely unstable with a wide range of variation both in time and across cells. It is shown that conventional scheduling came short to provide a sustainable cell throughput and the unstable throughput performance was spread over all cells. It is also shown that throughput can and will be stabilized as direct result of multi-cell coordination of intercell interference control by implementing Load Matrix technique. Comparing with benchmark scheduler, when LM scheduler was implemented there was no longer “best” and “worst” cell in terms of cell throughput as LM uniformly stabilized throughput for all cells. In other words, Load Matrix has not only increased cell throughput substantially, but also stabilized that with a high level of consistency.

9.2 Future Work

The concept of base station coordinated scheduling and intercell interference control has been put to live field test in Autumn 2009 as part of LTE-Advanced [41] study under generic name Coordinated Multi-Point transmission (CoMP) [42]. With SC-FDMA scheme employed at physical layer in uplink LTE, the major performance limiting factor specifically at the cell-edge
was due to heavy inter-cell interference. In order to improve this and thereby enhance the cell-edge performance, CoMP has been investigated in which the transmission and/or reception was coordinated over multiple cells. The uplink control signalling from a UE can be received by multiple cells but scheduling still took place at UE’s serving cell [43]. Field trial results showed that CoMP will make it possible for mobile users to enjoy consistent performance and quality when they use high-bandwidth applications regardless of their position in the cell. Even at the cell edge where transmission quality is typically poor and difficult to maintain, data rates greater than 5Mbps were reported for a vast majority of locations. Tight coordination of the transmission and reception of signals at multiple access points proved to be very effective in reducing interference and increasing efficiency [44]. LM technique at system level combined with MIMO at link level is a promising solution candidate to combine with CoMP in LTE-Advanced system. Further study is required however to come up with the best combined LM-CoMP solution.

When LM is jointly deployed with CoMP, the RoT target can be set higher as some of the inter-cell load elements participated in cell RoT is no longer interference but useful signal. CoMP will increase overall cell throughput while LM maintains the interference outage and provides performance uniformity and longevity. In addition, configuration and coordination of the cooperating cells for each UE in LM needs to be studied further taking into account the requirements of CoMP technique. It is desirable to optimize CSI feedback in a way to capture both LM and CoMP requirements and hence minimize signalling overhead. One possibility is to identify shared feedback information that can be used by both techniques and avoid duplication in signalling. Nevertheless, evaluation of overall LM-CoMP practical performance gain versus complexity is essential. For OFDMA-based systems, LM technique can still be used but definition of load elements and RoT needs to be clarified. For example, one can define “effective RoT” to measure load in a similar approach as “effective SINR” is defined in multi-carrier OFDMA systems. The closed-form definition of load element and RoT in these systems remains for further study.

As stated before, our focus in this thesis was on uplink only. LM technique is developed primarily for uplink scheduling and interference management, and it is not applicable to downlink without necessary adaptations. In downlink, Cell edge users become the main measuring point for intercell interference (as oppose to base stations in uplink). Participating load elements at each cell edge user come from a handful of neighbouring base stations. Expanding LM technique to downlink scheduling and intercell interference control is an interesting challenge for future work.
Publications


Award

Bibliography


Appendix A: Simulation platform

A.1 Introduction

In chapter 2, a number of scheduling techniques for uplink have been presented and discussed including Nokia's RBF-based decentralised/NodeB scheduling algorithm, Qualcomm's centralised/RNC and decentralised/NodeB scheduling algorithm and Fujitsu's VCUP. As explained there, VCUP is a fine-tuning algorithm introduced for TFC selection at UE. It takes into account the competition in a cell by means of a Comparative Metric (CM), which could be for instance, the buffer status information of UEs. In order to draw conclusion without drifting from the main point of argument, only a limited selection of simulation results have been presented in chapter 4 and summary of observations provided.

In this appendix, for anyone who is interested to have further insight into the simulation cases and results, we present wider range of simulations undertaken with different load conditions and network settings.

First, VCUP performance is evaluated against other benchmark algorithms followed by further optimisations and enhancement. Then, two novel distributed algorithms proposed in chapter 3 namely UniS Algo-1 & Algo-2 are examined in more details.

Numerous simulation runs have been carried out to evaluate and compare the performance with different system loads and with hot spots.

Finally, detailed description on the simulations undertaken is presented, including simulation scenarios, parameters, and a complete set of simulation results.

A.2 Evaluation summary of VCUP algorithm

A.2.1 CM2 (VCUP1)

The main objective here was to construct a full-scale system level simulator for UMTS enhanced uplink FDD packet transmission (HSUPA) and to evaluate and identify the gain of Fujitsu's VCUP algorithm presented in chapter 2 against the benchmark algorithms namely Nokia and Qualcomm schedulers.
First, a system level simulation platform has been developed and validated. From the validation results presented, the performance of the platform fits very well with the 3GPP release-99 system.

Two benchmark scheduling algorithms, Nokia’s and Qualcomm’s decentralised scheduling, are then implemented and tested. Together with VCUP algorithm, they have been evaluated thoroughly. The following cases of VCUP have been defined:

- **VCUP Case A**: VCUP is executed after TFC selection based on NodeB assigned rate, UE available power, and buffer occupancy, hence VCUP rate is limited by NodeB assigned rate, UE available power, and buffer occupancy.
- **VCUP Case B**: VCUP is executed after TFC selection based on UE available power, and buffer occupancy, hence VCUP rate is limited by UE available power, and buffer occupancy, but is not limited by NodeB assigned rate.
- **VCUP Case C**: VCUP is executed after TFC selection based on NodeB assigned rate, UE available power, hence VCUP rate is limited by NodeB assigned rate and UE available power, but is not limited by buffer occupancy.
- **VCUP Case D**: VCUP is executed after TFC selection based on UE available power, hence VCUP rate is limited by UE available power, but is not limited by NodeB assigned rate or buffer occupancy.

After intensive rounds of simulations, the following are the main observations:

- Qualcomm Node B scheduler shows better performance in terms of TFC utilization, total queue size and packet delay compared to Nokia’s Node B scheduler.
- The average RoT is higher in case of Nokia’s Node B scheduler with VCUP compared to Nokia’s Node B scheduler without VCUP.
- The average RoT is almost similar in case of Qualcomm Node B scheduler with VCUP compared to Qualcomm Node B scheduler without VCUP.
- The TFC utilization is better when VCUP is being used for both Nokia’s Node B scheduler and Qualcomm Node B scheduler.
- VCUP A and C increase the amount of system total queue size in both Nokia and Qualcomm cases while VCUP B and D decrease the amount of system total queue size in both Nokia and Qualcomm cases.
- VCUP A and C increase the packet delay in both Nokia and Qualcomm cases while VCUP B and D decrease the packet delay in both Nokia and Qualcomm cases.
• VCUP A and C decrease both OTA and service throughput in both Nokia and Qualcomm cases.
• VCUP B and D have slightly higher OTA throughput in Qualcomm case and same OTA throughput in Nokia case but their service throughput are worse. It could be because they cannot take care of cell congestion properly therefore the packet error probability would be higher.

It has been observed that VCUP case B outperforms other cases when working together with Nokia or Qualcomm Node-B scheduler. For simplicity, VCUP case B is renamed as VCUP1.

A.2.2 CM3 (VCUP2)

The main objective here was to further optimise and improve VCUP1 algorithm, and investigate the possible enhancements to it for better performance. To achieve this goal, we investigate and evaluate the performance of VCUP (CM3) case B, which is renamed as VCUP2.

CM3 is the Comparative Metric enhanced by taking into account user’s channel condition. In other words, CM3 is based on buffer status as in CM2 defined in (2.14), as well as user’s channel condition (SIR). Let us repeat (2.14) here as (A.1) which captures the effect of buffer occupancy in CM and take it from there.

\[
C_{nj}(m) = CM_{nt} \cdot CM_{nt2j}, \quad n=1...N_j
\]  

(A.1)

Now, Node-B divides UE’s SIR value by the maximum SIR (not in dB) among all the UEs so that

\[
Norm_{SIR_{nj}}(m) = \frac{SIR_{nj}(m)}{SIR_{max,j}}, \quad n=1...N_j
\]

(A.2)

where index \(m\) represents current TTI or uplink scheduling event, \(N\) is the total number of source UEs, \(SIR_{max,j}\) is the maximum SIR and depends on the service group \(j\).

Node-B for each UE multiplies this value \(Norm_{SIR_{nj}}(m)\) by 100, takes the integer part and sends the result, which represents a ratio between zero and one. Then determines the distance from minimum normalized SIR for each UE

\[
Dist_{min \_ SIR_{nj}}(m) = Norm_{SIR_{nj}}(m) - Norm_{SIR_{min,j}}(m), \quad n=1...N_j
\]

(A.3)

Node-B then normalizes this value to the sum of distances so that

\[
Dist_{\_ min \_ SIR_{nj,norm}}(m) = \frac{Dist_{min \_ SIR_{nj}}(m)}{\sum_{j=1}^{N_j} Dist_{min \_ SIR_{nj}}(m)}, \quad n=1...N_j
\]

(A.4)

In a similar way, distance from average SIR is then determined in Node-B.
Appendix A

\[
\text{Dist}_{\text{avg}} \text{ SIR}_{n,j}(m) = \text{Norm}_{\text{SIR}}_{n,j}(m) - \text{Norm}_{\text{SIR avg }}_{j}(m), \quad n=1\ldots N_j
\]  
(A.5)

Since the value of \( \text{Dist}_{\text{avg}} \text{ SIR}_{n,j}(m) \) might be negative between -0.99 and +0.99, a positive bias value is added so that

\[
\text{Dist}_{\text{avg}} \text{ SIR}_{n,j}(m) = \text{Dist}_{\text{avg}} \text{ SIR}_{n,j}(m) + 1.0, \quad n=1\ldots N_j
\]  
(A.6)

then

\[
\text{Dist}_{\text{avg}} \text{ SIR}_{n,j,\text{norm}}(m) = \text{Dist}_{\text{avg}} \text{ SIR}_{n,j}(m) \sum_{n=1}^{N_j} \text{Dist}_{\text{avg}} \text{ SIR}_{n,j}(m), \quad n=1\ldots N_j
\]  
(A.7)

Let us define \( CM_{n,j} = \text{Dist}_{\text{min}} \text{ SIR}_{n,j,\text{norm}}(m) \) and \( CM_{n,j} = \text{Dist}_{\text{avg}} \text{ SIR}_{n,j,\text{norm}}(m) \).

\( CM_{n,j} \) and \( CM_{n,j} \) are then combined with \( CM_{n,j} \) and \( CM_{n,j} \) from (A.1):

\[
C_{n,j}(m) = CM_{n,j}CM_{n,j}CM_{n,j}CM_{n,j}, \quad n=1\ldots N_j
\]  
(A.8)

Node-B then normalises this product to the maximum combined value so that

\[
CM_{n,j}(m) = C_{n,j}/C_{n,j,\text{max}}, \quad n=1\ldots N_j
\]  
(A.9)

which is the definition of \( CM_3 \).

After several rounds of simulations, the following conclusions have been drawn from the results:

- The best server selection scheme based on uplink SIR results in better performance for Nokia and Qualcomm Node-B schedulers. However, it does not bring gains to the VCUPl performance.
- SIR-assisted VCUPl does not bring any gain against VCUPl. In fact, it performs worse than VCUPl.
A.3 Hotspot Modelling

An enhanced system level simulator is used, modelling cell layout with hot spots as shown in Figure A-1. There are three hot spots, located in the three sectors of the central NodeB. These hot spot cells have more users per cell. For instance, we could have 30 UEs per cell for hot spot cells whereas 10 UEs per cell for normal cells. In the simulator, only the initial number of users in hot spots or normal cells is controlled, and there is no limitation applied on the mobility between hot spot cells and normal cells. Hence the number of users in hot spot cells could change when users move around during the simulation.

![Cell layout with hot spots](image)

**Figure A-1: Cell layout with hot spots**

Two basic test cases are defined for the simulations with hot spots: hot spot case A and hot spot case B. In hot spot case A, there are 15 UEs per cell for hot spot cells and 10 UEs per cell for normal cells; whereas in hot spot case B, there are 30 UEs per cell for hot spot cells and 10 UEs per cell for normal cells. An example of user distribution for hot spot case B is shown in Figure A-2.
Figure A-2: User distribution (10 UEs per cell, hot spot case B)

A.4 System level Simulations

30 simulation cases have been considered as follows:

1) Nokia scheduler, 10 UEs per cell
2) Nokia scheduler, 15 UEs per cell
3) Nokia scheduler with hot spots, case A
4) Nokia scheduler with hot spots, case B
5) Qualcomm scheduler, 10 UEs per cell
6) Qualcomm scheduler, 15 UEs per cell
7) Qualcomm scheduler with hot spots, case A
8) Qualcomm scheduler with hot spots, case B
9) UniS algo-l + CM2 without SHO combining, 10 UEs per cell
10) UniS algo-l + CM2 without SHO combining, 15 UEs per cell
11) UniS algo-l + CM2 with SHO combining with hot spots, case A
12) UniS algo-l + CM2 with SHO combining with hot spots, case B
13) UniS algo-l + CM2 with SHO combining, 10 UEs per cell
14) UniS algo-l + CM2 with SHO combining, 15 UEs per cell
15) UniS algo-l + CM2 with SHO combining with hot spots, case A
16) UniS algo-l + CM2 with SHO combining with hot spots, case B
17) UniS algo-1 + CM3 without SHO combining, 10 UEs per cell
18) UniS algo-1 + CM3 without SHO combining, 15 UEs per cell
19) UniS algo-1 + CM3 without SHO combining with hot spots, case A
20) UniS algo-1 + CM3 without SHO combining with hot spots, case B
21) UniS algo-1 + CM3 with SHO combining, 10 UEs per cell
22) UniS algo-1 + CM3 with SHO combining, 15 UEs per cell
23) UniS algo-1 + CM3 with SHO combining with hot spots, case A
24) UniS algo-1 + CM3 with SHO combining with hot spots, case B
25) CM2 without SHO combining, 10 UEs per cell
26) CM2 without SHO combining, 15 UEs per cell
27) CM2 with SHO combining, 10 UEs per cell
28) CM2 with SHO combining, 15 UEs per cell
29) UniS algo-2, 10 UEs per cell
30) UniS algo-2, 15 UEs per cell

It is worth to remind that UniS algo-1 is based on CI (cell capacity indicator as explained in section 3.2) as well as CM where as UniS algo-2 is based on CI and user queue size. Also to note, CM2 represents Comparative Metric in VCUP algorithm, which is only based on UE’s buffer status as defined in (2.15) whereas CM3 is the Comparative Metric based on both buffer status and channel condition (SIR).

In all these cases, maximum cell load threshold is set to 0.7 (5.3 dB in RoT) and the best server selection is based on uplink SIR. Complete set of simulation parameters is presented in Table A.1.
### Table A.1: Simulation Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Explanation</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cellular layout</td>
<td>Hexagonal grid, 3-sector sites</td>
<td></td>
</tr>
<tr>
<td>Site to Site distance</td>
<td>2800 m</td>
<td></td>
</tr>
<tr>
<td>Antenna pattern</td>
<td>0 degree horizontal azimuth is East 70 degree (-3dB), 20dB front-to-back ratio</td>
<td></td>
</tr>
<tr>
<td>Propagation model</td>
<td>$L = 128.1 + 37.6 \log_{10}(R)$ R in kilometres</td>
<td></td>
</tr>
<tr>
<td>Std. deviation of slow fading</td>
<td>8.0 dB</td>
<td>Log-Normal Shadowing</td>
</tr>
<tr>
<td>Carrier frequency</td>
<td>2000 MHz</td>
<td></td>
</tr>
<tr>
<td>Node B antenna gain plus Cable Loss</td>
<td>14 dBi</td>
<td></td>
</tr>
<tr>
<td>Node B RX diversity</td>
<td>Uncorrelated 2-antenna RX diversity</td>
<td></td>
</tr>
<tr>
<td>UE antenna gain</td>
<td>0 dBi</td>
<td></td>
</tr>
<tr>
<td>Maximum UE EIRP</td>
<td>21 dBm</td>
<td></td>
</tr>
<tr>
<td>BS total Tx power</td>
<td>43 dBm</td>
<td>Relative to the maximum power</td>
</tr>
<tr>
<td>Downlink CPICH power</td>
<td>-10 dB</td>
<td>Relative to the maximum power</td>
</tr>
<tr>
<td>Other downlink common channels</td>
<td>-10 dB</td>
<td></td>
</tr>
<tr>
<td>Uplink system noise</td>
<td>-102.9 dB</td>
<td></td>
</tr>
<tr>
<td>Soft Handover Parameters</td>
<td>Window_add = 4 dB, Window_drop = 6 dB</td>
<td>Window_add: The signal from a BS has to be at highest this amount smaller than the current active set's best BS's signal for a BS to be added in the active set. Window_drop: When the signal from a BS has dropped below the active set's best BS's signal minus this parameter, the BS will be dropped from the active set.</td>
</tr>
<tr>
<td>Fast Fading model</td>
<td>Jakes spectrum, Doppler based on speed</td>
<td></td>
</tr>
<tr>
<td>Uplink Power Control</td>
<td>Closed-loop power control delay: one slot</td>
<td>Power control feedback: BER = 4% for a Node-B-UE pair.</td>
</tr>
<tr>
<td>Power Control Step</td>
<td>1 dB</td>
<td></td>
</tr>
<tr>
<td>Power Control Delay</td>
<td>1 Time Slot</td>
<td></td>
</tr>
<tr>
<td>User data rates in TFCS allocated to the UE</td>
<td>TFCS1: 8, 16, 32, 64, 128, 256, 384 kbit/s</td>
<td></td>
</tr>
<tr>
<td>TTI</td>
<td>10 ms</td>
<td></td>
</tr>
<tr>
<td>Scheduler period</td>
<td>100 ms</td>
<td></td>
</tr>
<tr>
<td>Noise Rise Target (NRT)</td>
<td>5.2 dB</td>
<td>Same for all simulations</td>
</tr>
<tr>
<td>Priority</td>
<td>Proportional Fair</td>
<td></td>
</tr>
<tr>
<td>Minimum allowed data rate</td>
<td>6 kbps</td>
<td></td>
</tr>
<tr>
<td>Maximum scheduled data rate</td>
<td>384 kbps</td>
<td></td>
</tr>
<tr>
<td>Maximum UE Buffer size</td>
<td>384000 bits</td>
<td></td>
</tr>
<tr>
<td>Traffic model</td>
<td>Gaming</td>
<td>Source traffic rate 115 kbps</td>
</tr>
</tbody>
</table>
A.4.1 Simulation Results for Cases without Hot Spots

A.4.1.1 Cell Load: 10 UEs per Cell

Figure A-3: PDF of RoT (10 UEs per cell)

Figure A-4: CDF of RoT (10 UEs per cell)
Figure A-5: PDF of total queue size (10 UEs per cell)

Figure A-6: CDF of total queue size (10 UEs per cell)
Figure A-7: PDF of packet delay (10 UEs per cell)

Figure A-8: CDF of packet delay (10 UEs per cell)
Figure A-9: OTA throughput vs. distance from cell site (10 UEs per cell)

Figure A-10: Service throughput vs. distance from cell site (10 UEs per cell)
Appendix A

Figure A-11: Buffer occupancy vs. distance from cell site (10 UEs per cell)

Figure A-12: Mean packet delay vs. distance from cell site (10 UEs per cell)
A.4.1.2 Cell Load: 15 UEs per Cell

PDF of RoT (15 UEs per cell)

![PDF graph showing different scheduler performances](image)

- **Mean:**
  - NOKIA Scheduler: 2.2901
  - Qualcomm Scheduler: 2.4292
  - CM2 without SHO-Comb: 3.0334
  - CM2 with SHO-Comb: 3.1224
  - UniS Alg1+ CM2 without SHO-Comb: 4.1602
  - UniS Alg1+ CM2 with SHO-Comb: 4.6067
  - UniS Alg1+ CM3 without SHO-Comb: 2.6779
  - UniS Alg1+ CM3 with SHO-Comb: 2.6918
  - UniS Alg2: 3.2388

- **Std:**
  - NOKIA Scheduler: 0.06237
  - Qualcomm Scheduler: 1.0363
  - CM2 without SHO-Comb: 1.9776
  - CM2 with SHO-Comb: 1.489
  - UniS Alg1+ CM2 without SHO-Comb: 2.6791
  - UniS Alg1+ CM2 with SHO-Comb: 2.5262
  - UniS Alg1+ CM3 without SHO-Comb: 1.746
  - UniS Alg1+ CM3 with SHO-Comb: 1.6991
  - UniS Alg2: 1.9973

Figure A-13: PDF of RoT (15 UEs per cell)

CDF of RoT (15 UEs per cell)

![CDF graph showing different scheduler performances](image)

Figure A-14: CDF of RoT (15 UEs per cell)
Figure A-15: PDF of total queue size (15 UEs per cell)

Figure A-16: CDF of total queue size (15 UEs per cell)
Appendix A

Figure A-17: PDF of packet delay (15 UEs per cell)

Figure A-18: CDF of packet delay (15 UEs per cell)
Figure A-19: OTA throughput vs. distance from cell site (15 UEs per cell)

Figure A-20: Service throughput vs. distance from cell site (15 UEs per cell)
Figure A-21: Buffer occupancy vs. distance from cell site (15 UEs per cell)

Figure A-22: Mean packet delay vs. distance from cell site (15 UEs per cell)
A.4.2 Simulation Results for Cases with Hot Spots

A.4.2.1 Hot Spot Simulation Case A

Figure A-23: PDF of RoT (10 UEs per cell, hot spot case A)

Figure A-24: CDF of RoT (10 UEs per cell, hot spot case A)
Figure A-25: PDF of total queue size (10 UEs per cell, hot spot case A)

Figure A-26: CDF of total queue size (10 UEs per cell, hot spot case A)
Figure A-27: PDF of packet delay (10 UEs per cell, hot spot case A)

Figure A-28: CDF of packet delay (10 UEs per cell, hot spot case A)
Appendix A

**Figure A-29:** OTA throughput vs. distance from cell site (10 UEs per cell, hot spot case A)

**Figure A-30:** Service throughput vs. distance from cell site (10 UEs per cell, hot spot case A)
Figure A-31: Buffer occupancy vs. distance from cell site (10 UEs per cell, hot spot case A)

Figure A-32: Mean packet delay vs. distance from cell site (10 UEs per cell, hot spot case A)
Table A.2: Results summary for hot spot performance (case A)

<table>
<thead>
<tr>
<th></th>
<th>Mean ROT (dB)</th>
<th>Cell OTA Throughput (Mbps)</th>
<th>Cell Service Throughput (bps)</th>
<th>Mean Packet Delay (TTI)</th>
</tr>
</thead>
<tbody>
<tr>
<td>NOKIA scheduler</td>
<td>1.9359</td>
<td>1.1297</td>
<td>1.0885</td>
<td>14.0181</td>
</tr>
<tr>
<td>Qualcomm scheduler</td>
<td>1.9011</td>
<td>1.1252</td>
<td>1.1029</td>
<td>16.3344</td>
</tr>
<tr>
<td>UniS Algo-1 + CM2 without SHO-Comb</td>
<td>2.7702</td>
<td>1.2727</td>
<td>1.1146</td>
<td>9.0262</td>
</tr>
<tr>
<td>UniS Algo-1 + CM2 with SHO-Comb</td>
<td>2.6947</td>
<td>1.2130</td>
<td>1.1220</td>
<td>7.6761</td>
</tr>
<tr>
<td>UniS Algo-1 + CM3 without SHO-Comb</td>
<td>2.2588</td>
<td>1.1932</td>
<td>1.1000</td>
<td>17.6478</td>
</tr>
<tr>
<td>UniS Algo-1 + CM3 with SHO-Comb</td>
<td>2.5905</td>
<td>1.1658</td>
<td>1.0852</td>
<td>26.7275</td>
</tr>
</tbody>
</table>

A.4.2.2 Hot Spot Simulation Case B

Figure A-33: PDF of RoT (10 UEs per cell, hot spot case B)
Appendix A

Figure A-34: CDF of RoT (10UEs per cell, hot spot case B)

Figure A-35: PDF of total queue size (10 UEs per cell, hot spot case B)
Figure A-36: CDF of total queue size (10 UEs per cell, hot spot case B)

Figure A-37: PDF of packet delay (10 UEs per cell, hot spot case B)
Figure A-38: CDF of packet delay (10 UEs per cell, hot spot case B)

Figure A-39: OTA throughput vs. distance from cell site (10 UEs per cell, hot spot case B)
Figure A-40: Service throughput vs. distance from cell site (10 UEs per cell, hot spot case B)

Figure A-41: Buffer occupancy vs. distance from cell site (10 UEs per cell, hot spot case B)
Appendix A

Mean packet delay vs. distance from cell site (10 UEs per cell)

Figure A-42: Mean packet delay vs. distance from cell site (10 UEs per cell, hot spot case B)

Table A.3: Results summary for hot spot performance (case B)

<table>
<thead>
<tr>
<th></th>
<th>Mean ROT (dB)</th>
<th>Cell OTA Throughput (Mbps)</th>
<th>Cell Service Throughput (bps)</th>
<th>Mean Packet Delay (TTI)</th>
</tr>
</thead>
<tbody>
<tr>
<td>NOKIA scheduler</td>
<td>2.9636</td>
<td>2.1925</td>
<td>2.1371</td>
<td>28.8344</td>
</tr>
<tr>
<td>Qualcomm scheduler</td>
<td>3.3976</td>
<td>2.5095</td>
<td>2.4611</td>
<td>18.0771</td>
</tr>
<tr>
<td>UniS Algo-1 + CM2 without SHO-Comb</td>
<td>6.4959</td>
<td>2.8041</td>
<td>2.5293</td>
<td>37.1602</td>
</tr>
<tr>
<td>UniS Algo-1 + CM2 with SHO-Comb</td>
<td>6.4425</td>
<td>2.7655</td>
<td>2.4829</td>
<td>45.8548</td>
</tr>
<tr>
<td>UniS Algo-1 + CM3 without SHO-Comb</td>
<td>3.7007</td>
<td>2.2578</td>
<td>2.0976</td>
<td>70.1380</td>
</tr>
<tr>
<td>UniS Algo-1 + CM3 with SHO-Comb</td>
<td>3.3697</td>
<td>2.1103</td>
<td>1.9531</td>
<td>80.1442</td>
</tr>
</tbody>
</table>
A.4.3 Summary

Following simulations have been performed:

- Simulations have been carried out to evaluate and compare the performance of Nokia’s RUF-based scheduler, Qualcomm’s NodeB scheduler, VCUP, and UniS distributed schedulers Algo-1 & Algo-2 with different system loads.

- Simulations have been carried out to evaluate and compare the performance of Nokia’s RUF-based scheduler, Qualcomm’s NodeB scheduler, VCUP, UniS distributed schedulers Algo-1 & Algo-2 with hot spots.

From simulation results provided, some important conclusions can be drawn:

- When the system loading is 10 UEs per cell, it is observed that UniS Algo-1 with CM2 could achieve superior performance compared against all the other type of algorithms, in terms of throughput, packet delay, buffer occupancy etc.

- When the system loading increases to 15 UEs per cell, it is observed that UniS Algo-1 with CM2 could still achieve superior performance in general, but the gains decrease compared with benchmark algorithms, such as Qualcomm or Nokia scheduling algorithm.

- In the simulations with hot spots, similarly, UniS Algo-1 with CM2 outperforms all other algorithms in terms of throughput, buffer occupancy etc., however when the system loading increases, the gains decrease.

- It is observed that for those users close to the cell site, UniS Algo-1 with CM3 could achieve very good performance in terms of throughput, packet delay, buffer occupancy etc., but not for those users who are far away. This is due to the fact that CM3 takes the channel condition into account during rate selection.

- CM combination with soft handover (SHO) does not show any gain, in comparison with CM without soft handover case.